DESIGN OF GAME THEORETIC CONTROLLERS FOR MULTIOBJECTIVE LINEAR REGULATOR PROBLEMS

BY

JIANN-WOEI JANG

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

1995

Dedicated to

my parents,
who always loved;
my wife,
who always encouraged;
my son,
who always delighted.

ACKNOWLEDGEMENTS

The author would like to express his sincere gratitude to his advisor, Dr. Norman Fitz-Coy, for guidance, encouragement, support, and friendship. Dr. Fitz-Coy has played a major role in providing the author with a solid background in the author's chosen field of research. The author is extremely grateful for all Dr. Fitz-Coy's unconstrained help.

The author also wishes to express his gratitude to Dr. Haniph Latchman, the external member in his committee, for his invaluable advice and guidance. The author also wants to thank the rest of the members of his committee, Drs. Bernard Mair, Ulrich Kurzweg and Robert Hirko, for their helpful advices.

Finally, the author would like to express his heartfelt thanks to his parents and wife for all their unconditioned love and support. His academic improvements could not have been achieved without their endless encouragement.

TABLE OF CONTENTS

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

DESIGN OF GAME THEORETIC CONTROLLERS FOR MULTIOBJECTIVE
LINEAR REGULATOR PROBLEMS

By

Jiann-Woei Jang

December 1995

Chairman: Dr. Norman Fitz-Coy
Major Department: Aerospace Engineering, Mechanics and Engineering Science

This dissertation addresses the issue of applying game theoretic controllers to the multiobjective linear regulator problems. The classical regulator problem in optimal control theory is based on the assumption that feedback controllers optimize a linear system with a single integral performance index. For the general multiple input/multiple output linear system, there may exist more than one performance index. Differential game theory, which frequently deals with multiple performance indices, is found to be directly applicable for the multiobjective control problems. This observation motivates this research to design game theoretic controllers for the multiobjective linear regulator problems.

There are two ways to design game theoretic controllers in the multiobjective linear regulator problem. For a given set of weighting matrices associated with the integral performance indices, the feedback gain matrix can be obtained by solving coupled algebraic Riccati equations. Most up-to-date coupled algebraic Riccati equation solvers are based on an iterative Newton method. It is well known that convergence for such an iterative scheme strongly depends on the initial guesses. In the first part of this research, a robust integrative

scheme based on homotopy theory is developed. Current homotopy schemes can generate solutions for algebraic Riccati-type equations; however, the solutions are not guaranteed to be positive definite. For a standard algebraic Riccati equation, the proposed homotopy scheme generates the positive definite solution, which is the only solution of interest in control problems. To make the homotopy scheme more efficient, the symmetric property of the solution of an algebraic Riccati-type equation is used in conjunction with the Kronecker sum.

In the second part of this research, a new optimal pole placement scheme based on the differential game theory is developed for a linear time invariant system with prescribed closed-loop pole locations. The pole placement scheme is capable of shifting a single pole or a pair of poles to the desired location(s), while providing "additional" design specifications for the linear system. Multiple pole shifting tasks can be implemented by utilizing the shifting algorithm for a single pole or a pair of poles recursively. It is demonstrated that the proposed scheme expands the design domain from classical optimal control scenario and allows additional design criteria to be imposed on the closed-loop system.

CHAPTER 1
INTRODUCTION

1.1 Motivation

The classical regulator problem in optimal control theory is based on the assumption that state feedback controllers optimize the performance of a linear system based on a single integral quadratic performance index. However, in many real dynamical problems, it may not be sufficient to describe the multiple objectives of a multiple input/multiple output linear system with a single integral quadratic performance index [1,2]. For example, consider the communication satellite shown in Fig. 1.1. On one hand, the "communication payload" is maneuvered to point to earth for the purpose of data transmission. On the other hand, the "energy payload" must be oriented such that the on-board solar panel can absorb the maximum amount of solar energy [1]. These are conflicting requirements which must be addressed by the on-board controllers. Another example of a multiple objective dynamical system is the United States Microgravity Payload (USMP), which uses the unique attributes of the space shuttle cargo bay to conduct large, sophisticated microgravity experiments. In the USMP program, the space shuttle is maneuvered to accomplish its mission in the space. At the same time, the on-board microgravity payload must be isolated from the space shuttle's motion in order to perform dedicated scientific experiments such as crystal growth.

In general, the strategic interactions among all controllers for such multiobjective multiple input/multiple output systems cannot be properly described by the classical optimal control theory which only deals with a single optimization objective. For most dynamical systems, further analysis on the multiobjective linear and/or nonlinear system should be further developed. A differential game theory, which is the study of interacting strategies

for multiple intelligent players, is found to be directly applicable for these multiobjective control problems.

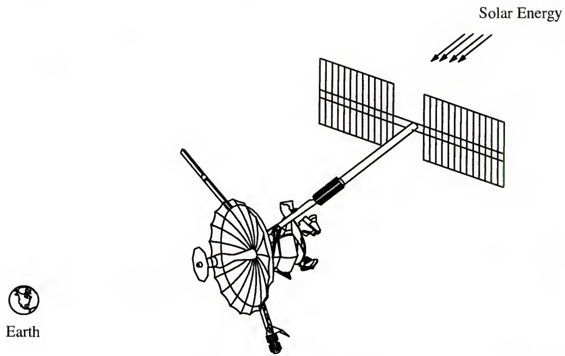

Figure 1.1 Communication Satellite

It has been decades since the initial model of the differential game in a linear system was proposed by Isaacs [3]. Starr and Ho [4] proposed a nonzero-sum noncooperative differential game with a given set of multiple integral performance indices for the general n-player game. The weighting matrices in those integral performance indices are chosen in advance. In the finite time case, the time varying solutions to an n-player nonzero-sum noncooperative differential game can be obtained by solving n-coupled differential Riccati equations. In the infinite time horizon case, the time invariant solution is obtained by solving n-coupled algebraic Riccati equations.

However, due to its mathematical complexity, the majority of current research has been restricted to the solution of a 2-player, nonzero-sum, noncooperative, infinite time horizon differential game. Most solution schemes currently in use are based on Newton's iterative method [1,5,6]. Another scheme, based on a conjugate gradient technique was formulated

by Innocenti and Schmidt [7]. However, the convergence to the desired solutions of all iterative schemes strongly depends on the choice of the initial guesses, even for the 2-player differential game.

Besides the numerical difficulty of solving the general n-player nonzero-sum, noncooperative differential game, the most ambiguous issue to a control engineer is the selection of weighting matrices in those performance indices. Different combinations of weighting matrices will lead to entirely different closed-loop pole locations; thus completely different performance for the closed-loop system responses. To the best of the author's knowledge, the issue of weighing matrices selection for game theoretic state feedback controllers with prescribed closed-loop poles has never been addressed in the literature.

1.2 Research Scope

In this research, two methodologies for designing linear time invarient, game theoretic, state feedback controllers in the regulator problem are discussed. First, for a given set of weighting matrices for integral quadratic performance indices, the state feedback gain matrix can be obtained by solving coupled algebraic Riccati equations. Most up-to-date coupled algebraic Riccati equations solvers are based on Newton's iterative method. It is well known that convergence for such iterative schemes strongly depend on the initial guesses. A bad choice of an initial guess may result in divergence for the iterative scheme. To overcome such numerical difficulties, an integrative scheme based on homotopy theory is proposed in the first part of this research.

A homotopy scheme is a path following algorithm, which "bends" a known solution into the desired solution [8]. The homotopy path is described by the homotopy differential equation. For an algebraic Riccati-type equation, the associated homotopy differential equation is in the form of a Lyapunov-type equation. The proposed homotopy scheme has two features:

(1). In comparison with generic homotopy codes, the developed homotopy algorithm is more efficient in solving coupled algebraic Riccati equations. This improvement in efficiency resulted from exploiting the symmetric property of coupled algebraic Riccati equations and the use of the Kronecker sum.

(2). For standard algebraic Riccati equations, the current homotopy schemes [9,10] may not result in positive definite solutions. A homotopy scheme is developed which guarantees that the solution is positive definite; this is the solution of interest in the linear quadratic regulator problems.

An inverse methodology for designing a linear time invariant state feedback controller is defined as follow: "given a set of closed-loop pole locations and input weighting matrices in the integral quadratic performance indices, determine the state weighting matrices in the integral quadratic performance indices and hence the feedback gain matrices." Current optimal pole placement approaches are based on the assumption that the state feedback controllers optimize a linear system with a single objective. However, a single integral performance index cannot reflect the dynamic interactions between controller groups. Therefore, a new optimal pole placement methodology based on the differential game theory for a general multiobjective regulator problem is proposed in the second part of this research. By using multiple integral quadratic performance indices, the control domain can be expanded, thus allowing the system control designer the freedom of implementing additional criteria to specify the system's performance. The proposed method is capable of shifting either a single pole or a pair of poles (two real poles or a pair of complex conjugate poles) to the desired closed-loop location(s) while satisfying additional design specifications. Multiple pole shifting tasks can be implemented by utilizing the shifting algorithm from a single pole or a pair of poles recursively. It is found there are several advantages to the proposed methodology,

(1). The proposed methodology can determine state feedback controllers in an n-player differential game without actually solving the n-couple algebraic Riccati equations numerically.

(2). The proposed method can incorporate a design specification (e.g., minimize the envelop bound of full-order state time response) as an optimization criterion.

(3). The inverse LQR solution belongs to the solution set of the proposed optimization problem.

(4). The proposed methodology can incorporate other controller design considerations (e.g., constraints on the elements of the state feedback gain matrices) as additional equality or inequality constraints for this optimization problem.

1.3 Dissertation Outline

This dissertation is organized as follows. In Chapter 2, an introductory review of differential game theory is provided. The mathematical model for a multiobjective linear quadratic regulator is also introduced. In Chapters 3 and 4, integrative schemes based on homotopy theory are proposed. The symmetric property of a Riccati matrix in conjunction with the Kronecker sum are used to make the proposed algorithms more efficient. A robust algorithm, which guarantees a positive definite solution of the algebraic Riccati equation without finding all the solution, is presented in Chapter 3. In Chapter 4, an efficient algorithm for solving coupled algebraic Riccati equations is presented. An example shows the proposed homotopy scheme can determine a solution, which is not solvable by the Newton's iterative scheme.

The issue of pole placement and weighing matrices selection for game theoretic state feedback controllers are discussed in Chapters 5 and 6. Chapter 5 presents a pole shifting scheme which can be implemented by iteratively performing a single pole shifting or a pair of poles shifting task is proposed. There exist infinite solutions to the game theoretic controller by utilization of the proposed scheme. For both cases, it can be shown that the

solution to the LQR controller (inverse optimal problem) belongs to the solution set of the game theoretic controller. In order to get a finite solution set, a "superimposed" optimization criterion is needed. Chapter 6 presents some additional optimization criteria from which the control engineers can choose. These optimization criteria are

(1). Minimize the square of the Frobenius norm of the reduced-order state feedback gain matrix.

(2). Minimize the envelop bound of reduced-order state time response.

(3). Minimize the envelop bound of full-order state time response.

The three minimization criteria are chosen individually as a superimposed optimization criterion along the prescribed closed-loop pole location to form the three optimization models. A lateral motion model of an F-4 aircraft and a longitudinal motion model of an AIRC aircraft are used as numerical examples to demonstrate the feasibility of the proposed optimization model. In Chapter 7, a conclusion of this research as well as suggestions for future work is provided.

CHAPTER 2
APPLICATION OF GAME THEORETICAL CONTROLLER TO MULTIOBJECTIVE LINEAR SYSTEM

2.1 Introduction

For many large scale systems, there frequently exist many design performance indices to be accomplished by system engineers [11]. Thus, the multiobjective optimization problems, which study a system with multiple objectives, have become a popular research topic. The study of multiobjective problem in optimal control theory falls into the scope of differential game theory, which is an outgrowth of game theory and optimal control theory.

In this chapter, an introduction to multiobjective optimization problems and the differential games is given. The mathematical models, based on the differential game for the multiobjective linear quadratic regulators, are also introduced.

2.1.1 Historical Background

Game theory was first developed to analyze economic and/or war situations involving the multiple objectives of several individuals. The first study of game theory was initiated by Nash [12], who was the recipient of the Nobel Prize for Economics in 1994. Isaacs [3] adopted the concepts from game theory in conjunction with optimal control theory and introduced the theory of the differential game and its application to multiobjective linear systems. The analysis from Isaacs's research was based on a dynamic programming approach. Later on, Berkowitz and Fleming [13] utilized the principles from calculus of variations to treat differential games with a single integral objective, which is the prototype of the modern differential game. At that time, the researchers concentrated on a single integral quadratic performance index in the differential game. For the multiple input control

problem, the single objective differential game can be classified into two categories, namely zero-sum differential games and nonzero-sum differential games. In a 2-player differential game, a zero-sum differential game happens when each set of controller groups have a direct conflict of interest; one group of controllers chooses the most favorable action to an objective, and the other group of controller chooses the most unfavorable action to the same objective. In a 2-player differential game, a nonzero-sum differential game happens when two sets of controller groups do not have completely conflicting interests. The classical linear quadratic regulator for a multiple input system, which has a scalar integral quadratic performance index to be minimized, can be categorized as a nonzero-sum "cooperative" game. It is because all controllers are in the same group to minimize the scalar objective criteria, which can be defined as the "weighted" sum of all the individual objectives associated with each controller. Therefore, a nonzero-sum "cooperative" game theoretical controller for a multiobjective linear system cannot be found unless information about the "degree" of cooperation (or conflict) between all the objectives is provided. A cooperative optimal strategy gives an optimistic solution by assuming some degree of cooperation exists among all the objectives [12]. Starr and Ho [4] proposed a nonzero-sum, noncooperative differential game that has multiple integral quadratic performance indices to be minimized. A nonzero-sum noncooperative game gives a conservative, sometimes pessimistic solution which is based on the assumption that "cooperation" does not exist among all the controllers.

The definition of the multiobjective optimization problems is stated as follows [14]: given an admissible control space, U, the behavior of a system is characterized by an n-dimensional vector (or n-player in a differential game), $u = [u_1, u_2, ...u_n]^T$, $u \in U$, and is measured by an m-dimensional vector objective function (or m-cost functions in a differential game) $J(u) = [J_1(u), J_2(u), ...J_m(u)]^T$, the components of $J(u)$ are real functions of u. It is required to determine a point (points) $u^o \in U$, which optimizes the values of the objective functions $J_1(u), J_2(u), ...J_m(u)$ in some "sense."

Solution methodologies for the multiobjective optimization problems can be roughly classified into the following categories [15]:

(i) Optimization of all system design criteria criteria simultaneously.

(ii) Determination of a set of noninferior points,

(iii) Optimization of a set of system design criteria hierarchically,

The method of <u>optimization of all system design criteria simultaneously</u> is a conservative strategy in the sense that it assumes that there is no cooperative situation among all multiple objectives. The parallel strategy in game theoretic is called "Nash strategy" or "Nash equilibrium." For a given admissible control space, U, a strategy set $u^* \in U$, where $u^* = \{u_1^*, u_2^*, ...u_n^*\}$, is called a Nash strategy set in U relative to $J(u)$, if for $i = 1, ...n$, $J_i(u_1^*, u_2^*, ...u_n^*) \leq J_i(u_1^*, ..., u_{i-1}^*, u_i, u_{i+1}^*, ..., u_n^*)$ [4]. In other words, a Nash strategy is an optimal strategy for each of the players based on the assumption that all of the other players are sticking to their Nash equilibrium. In a noncooperative game, where the cooperation among the decision makers is inadmissible or at least difficult to enforce, Nash strategy is a reliable solution. This solution is secured against any attempt by one player to unilaterally alter its strategy, since any player can only lose by deviating from the equilibrium solution. A Nash strategy is applicable in the dynamical system with conflicting objectives like guidance problems [16] and collision avoidance strategies in robot mechanism [17].

The largest collection of research work is devoted to the method of <u>determination of a set of noninferior points</u>. For a given admissible control space, U, a point $u^o \in U$ is called a noninferior point in U relative to $J(u)$, if among all $u^o \in U$, there does not exist a point u such that $J_i(u) \leq J_i(u^o)$, $i = 1, 2, ...k$, with at least one of the inequalities being strict [18]. The parallel strategy in game theoretic is called "Pareto strategy" or "Pareto optimum." Many algorithms have been developed to implement the Pareto strategy in multiobjective dynamical systems [11,19,20,21].

The general procedure to find a Pareto optimum can be implemented in two steps [11]. The first step is to determine the degree of compromise between the conflicting objectives in some numerical sense. For example, in the two objective optimization problem, if the two objectives are "equally" important, then two weights, $\alpha_1 = 0.5$ and $\alpha_2 = 0.5$, will be assigned to the associated objectives. Once an "acceptable" compromise is reached, the multiobjective optimization problem can be converted into a scalar (or single) objective optimization by summing all the multiplication of performance indices with their associated weights. For instance, the two objective optimization example is converted to finding the optimum of a single objective J, where $J = \alpha_1 J_1 + \alpha_2 J_2$. Notice that the relative importance of these performance indices is generally unknown until the system's best capabilities are achieved and the trade-offs between the performance indices reached. An important inference from the above example is that solutions from the classical single performance index regulator problem belong to the Pareto solution set, since weights associated with each performance index are pre-chosen. The major drawback of classical optimal control technique to the regulator problem is that the solution is only optimal to a chosen weighted sum of all performance indices, but not necessary to the system. In other words, there may exist another solution (e.g., the Pareto optimal solution), with its associated weighted sum of performance indices, which will result in a better system performance than the current chosen one. Thus, an "optimal" Pareto solution is the solution (in the Pareto solution set) associated with the "optimal" weighting coefficient set.

By using a Pareto strategy in a multiobjective optimization problem, the condition characterizing a compromised solution can be expressed in an analytic form such that the trade-off alternatives can be determined with conventional optimization techniques. This condition, which follows the ordinary meaning of "compromise," is that "no improvement for any performance index can be achieved except at the expense of a degradation in at least one of the other performance indices" [11].

The method of optimization of system design criteria hierarchically is based upon the introduction of a preference ordering to the set of optimization criteria such that the multiobjective optimization problem is implemented by optimizing a set of scalar criteria sequentially. The parallel methodology in game theory is called the "Stackelberg strategy" or "Stackelberg optimum." The Stackelberg strategy requires the performance indices ranked in their order of importance [22]. The optimal value of the primary index alone is found first. The optimal solution of the secondary performance index is then subject to the constraint that the value of the primary index is held within some prescribed bonds of its optimum. The third performance index is treated similarly, with the first two optima included as additional constraints and so on, until all the performance indices have been considered. One typical example of using the Stackelberg strategy in the multiobjective optimization problem is the leader-follower problem [23]. In a leader-follower problem, the follower's objectives serve as the first performance indices set to be optimized, since the leaders must make their move according to followers' decision, which is optimal to the followers.

Another important game theoretic strategy that falls into this category is the "Mini-Max strategy." The major difference between the Mini-Max strategy and the Stackelberg strategy is that the former one has only one performance index and the latter one has multiple performance indices in the differential games. For a Mini-Max strategy, when Player 1 believes that the other players play a Nash strategy, Player 1 should also play a Nash strategy to avoid deviating from his minimum. However, if Player 1 cannot be sure of how his rivals select their strategies, Player 1 may choose to minimize his cost against the worst possible set of strategies his opponents could choose [17]. Therefore, it is an excessively pessimistic strategy, since Player 1's rivals may also choose their Nash strategies. In a 2-player Mini-Max game, there is usually only one performance index for the two players, a minimizer and a maximizer. The optimal value for the maximizer is determined first. The resulting optimum to the maximizer will be treated as an additional constraint while the

minimizer finds his optimum. One application of Mini-Max strategy in a multiobjective optimal control example is that of a Mini-Max or predator-prey problem [17]. The predator, who wants to minimize the distance between predator and prey, serves as a minimizer. On the contrary, the prey, who wants to keep away from the predator as far as possible, serves as a maximizer.

However, the problem with the application of a Stackelberg strategy stems from the fact that sometimes it is very difficult to determine the order of importance for all optimization criteria. Therefore, the application of a Stackelberg strategy becomes less effective for practical problems because optimization with respect to the chosen "most" important criterion has already led to a unique optimal solution and optimization of the other performance indices is only "sub-optimal" with respect to the first performance index. A good example is the predator-prey problem. If the predator's performance index is chosen as the first objective to be optimized, then the optimization result is always in favor of the predator, and vice versa. In general, the Stackelberg optimum may not produce acceptable results. For example, the prey may not want to follow the predator's optimum because the optimum is favorable to the opponent.

2.1.2 Comparison among Stackelberg, Pareto and Nash Strategies

An example of 2-player nonzero sum static/dynamic game [14,24,25] is shown in Fig. 2.1, where $u_1 u_2 \in R$, are feasible strategies for Player 1 (u_1) and Player 2 (u_2), and the cost functions $J_1(u_1, u_2)$ and $J_2(u_1, u_2)$ are convex and twice differentiable with respect to both arguments. Player 1 wants to minimize J_1 while Player 2 minimizes J_2. Equal cost contours in the space expanded by U_1 and U_2 are plotted for J_1 and J_2 in Fig. 2.1. Now, suppose Player 2 plays the game first by choosing u_{2_x}. Player 1 will choose u_{1_x} such that $J_1(u_{1_x}, u_{2_x}) \leq J_1(u_1, u_{2_x})$, where $u_1 \in U_1$. Player 1 can find u_{1_x} by drawing a horizontal line, $u = u_{2_x}$, which is tangent to an equal cost contour J_1 of Player 1. The U_1-coordinate of the tangent point is u_{1_x}, which minimize the cost J_1 when u_{2_x} is applied. Similarly, for

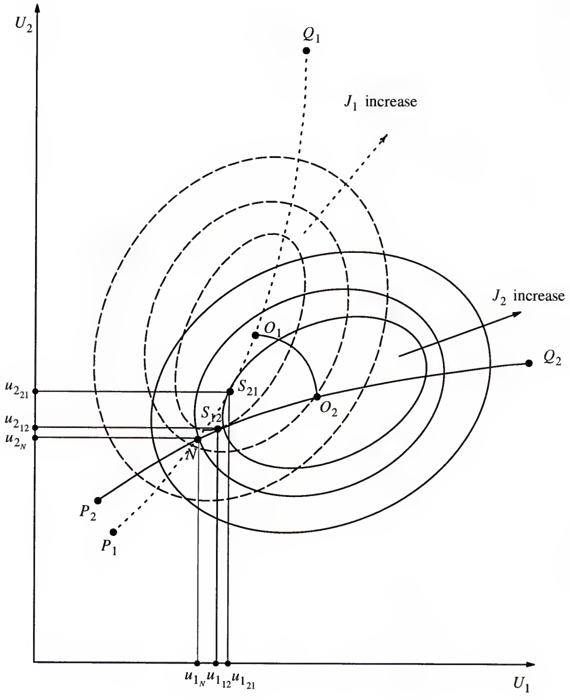

Figure 2.1 "N" is the location of the Nash strategy solution, "S_{12}" is the location of the Stackelberg strategy solution with Player 1 as a leader, "S_{21}" is the location of the Stackelberg strategy solution with Player 2 as a leader, and the reaction curve, O_1O_2, contains location of all possible Pareto strategy solutions.

the other choice of Player 2's strategy, Player 1 can find a "minimum" point. By repeating this procedure over the entire $U_1 \times U_2$ space, a reaction curve of Player 1, P_1Q_1, can be found. Correspondingly, if Player 1 plays first, then a reaction curve of Player 2, P_2Q_2, can be found.

In a Stackelberg game with Player 2 serving as a leader, Player 1 will make a move according to Player 1's reaction curve, P_1Q_1. Hence, Player 2 will choose his strategy by finding the minimum of J_2 along the reaction curve P_1Q_1. This can be achieved by finding a equal cost contour of J_2 which is tangent to P_1Q_1. Such a tangent point, S_{21}, as shown in Fig. 2.1, is a Stackelberg optimum with Player 2 serving as the leader. Likewise, S_{12} in Fig. 2.1 is the Stackelberg solution with Player 1 being the leader.

In a Nash game, the strategy pair (u_{1_N}, u_{2_N}) has the following properties that $J_1(u_{1_N}, u_{2_N}) \leq J_1(u_1, u_{2_N})$ and $J_2(u_{1_N}, u_{2_N}) \leq J_2(u_{1_N}, u_2)$ for any (u_1, u_2) in the $U_1 \times U_2$ space. The intersection point, N, of the reaction curves P_1Q_1 and P_2Q_2 gives the Nash equilibrium solution for the game.

The cost for the leader in a Stackelberg game is the best he can have for any pair of strategies on the reaction curve of the follower. So the cost for the leader is certainly at least as good as his cost of when a Nash strategy is used, provided that the follower plays along follower's reaction curve. In Fig. 2.1, both Stackelberg solutions are better than the Nash solution for both players. Situations where the follower will be worse off than when a Stackelberg strategy is played in a nonzero sum game also arise. However, if the leader does choose a strategy corresponding to a Stackelberg strategy, the follower will do worse by not following a Stackelberg solution himself.

In a Pareto game, the "scalar" objective function can be found by taking the weighted sum of all the objective functions. The Pareto solutions form the curve O_1O_2 by connecting all the tangent points of equal cost functions J_1 and equal cost functions J_2. Again, it is clear to see that classical optimal control is only one point on the O_1O_2 curve, since the single

performance index in the classical optimal control problem is the weighted sum of all the performance indices.

A Nash game is also entitled to a noncooperative game, since there is no cooperation between players. Similarly, a Pareto game is also called a cooperative game since "all" players agree to cooperate. Examples of the comparison between cooperative and noncooperative infinite time horizon differential games are made in References 4 and 26. It is often difficult to classify a game as either noncooperative or cooperative. In reality, the cooperation situation may change from time to time in the differential game [27]. If all the players benefit from cooperation in a time interval, then the Pareto solution is the optimal solution in the duration. However, if there exists one player who will not benefit from the cooperation, then the Nash solution is the reasonable solution in the time interval.

2.2 Multiobjective Linear Quadratic Regulator

For conciseness, the following historical review is based on a 2-objective 2-player differential game. Extension to higher order differential games follows directly. A linear dynamical system which mathematically models a controlled and observed system can be represented in the state space form,

$$\dot{x} = Ax(t) + B_1u_1(t) + B_2u_2(t) \tag{2.1}$$

$$y(t) = Cx(t)$$

where $u_1(t) \in \Re^{m_1}$ and $u_2(t) \in \Re^{m_2}$ represent separate control inputs to the plant (Player 1 and Player 2 of the game, respectively), and $y(t) \in \Re^k$ represents the plant outputs; $x(t) \in \Re^n$ represents the plant states (current condition of the game) and is assumed perfectly measured. The control actions u_1 and u_2 are restricted to state feedback (i.e., $u_i = u_i(x(t)),\ i = 1, 2$). Matrices A, B_1, B_2, and C are appropriately dimensioned, and time-invariant.

Application of game theoretic controllers with Nash, Pareto, Mini-Max and Stackelberg strategies to the multiobjective Linear Quadratic Regulator problems is discussed in the following sub-sections.

2.2.1 Game Theoretic Controller with Nash Strategy

A Nash game or noncooperative nonzero-sum differential game, is formulated by allowing the controllers (players) to separately minimize integral quadratic performance indices, J_1 and J_2, subject to the dynamical constraints imposed by Eq. (2.1). The solution is based on the assumption that the players know each other's performance indices and when the strategies have been calculated, they are announced at that time. Considering the infinite time horizon case, we define these integral quadratic performance indices as

$$J_1(u_1, u_2) = \int_0^\infty (x^T Q_1 x + u_1^T R_{11} u_1 + u_2^T R_{12} u_2)\, dt$$

$$J_2(u_1, u_2) = \int_0^\infty (x^T Q_2 x + u_1^T R_{21} u_1 + u_2^T R_{22} u_2)\, dt \tag{2.2}$$

where Q_1, Q_2, R_{12}, and R_{21} are positive semi-definite matrices, and R_{11}, R_{22} are strictly positive definite matrices [28].

The Nash control actions u_1^* and u_2^* are extremals of Eq. (2.2) and are subjected to the dynamical constraint equation (Eq. (2.1)), if they satisfy the following conditions:

$$J_1(u_1^*, u_2^*) \le J_1(u_1, u_2^*)$$

$$J_2(u_1^*, u_2^*) \le J_2(u_1^*, u_2) \tag{2.3}$$

The conditions in Eq. (2.3) are known as the Nash equilibrium conditions [1,4,6,28]. It can be shown that for these conditions to exist [28], the impulse response functions, $\partial H_i / \partial u_i$, $i = 1, 2$, and the influence functions, λ_i, $i = 1, 2$, must satisfy the following set of modified Euler-Lagrange equations

$$\left[\frac{\partial H_i}{\partial u_i}\right]_{u_1^*, u_2^*} = 0 \tag{2.4}$$

$$(\dot{\lambda}_i^*)^T = -\frac{\partial H_i}{\partial x} - \sum_{j=1, j\neq i}^{2} \frac{\partial H_i}{\partial u_j}\frac{\partial u_j}{\partial x} = 0;\ \lambda_i(t_0) = 0,\ i = 1, 2 \tag{2.5}$$

where H_i is the Hamiltonian associated with performance index J_i and is give by

$$H_i = x^T Q_i x + u_1^T R_{i1} u_1 + u_2^T R_{i2} u_2 + \lambda_i^T (Ax + B_1 u_1 + B_2 u_2) \tag{2.6}$$

The summation terms in Eq. (2.5) reflect the major difference between a noncooperative nonzero-sum differential game and any other differential games (e.g., noncooperative zero-sum games and cooperative games). These terms represent the effects changes in control strategy u_j would have on control strategies u_i; that is, the effects of controller cross-coupling. If each controller uses his state feedback strategy, then any perturbation due to one player will result in changes of all the co-state variables λ_i (through Eq. (2.5)) and hence changes of the state variables x. Consequently, all the controllers will be immediately influenced by the resulting perturbation of the state variables, since all the controller uses state feedback strategies. The explicit way of information exchange among all the controllers makes the noncooperative nonzero-sum differential game different from any other differential game, which uses an implicit pattern of information exchange (i.e., the information exchanges among all the state feedback controllers through their single performance index). This is the reason why the noncooperative nonzero-sum differential game is uniquely suited for problems involving conflicting situations among players (controllers).

If linear full-state feedback is assumed, the control strategies u_1^* and u_2^* satisfying Eq. (2.3) are [28]

$$u_i^* = -R_{ii}^{-1} B_i^T P_i x = -K_i^* x,\ i = 1, 2 \tag{2.7}$$

where $P_i,\ i = 1, 2$ are solutions of the coupled algebraic Riccati equations

$$- P_1A - A^TP_1 - Q_1 + P_1B_1R_{11}^{-1}B_1^TP_1 + P_1B_2R_{22}^{-1}B_2^TP_2$$
$$+ P_2B_2R_{22}^{-1}B_2^TP_1 - P_2B_2R_{22}^{-1}R_{12}R_{22}^{-1}B_2^TP_2 = 0 \tag{2.8}$$

$$- P_2A - A^TP_2 - Q_2 + P_2B_2R_{22}^{-1}B_2^TP_2 + P_2B_1R_{11}^{-1}B_1^TP_1$$
$$+ P_1B_1R_{11}^{-1}B_1^TP_2 - P_1B_1R_{11}^{-1}R_{21}R_{11}^{-1}B_1^TP_1 = 0 \tag{2.9}$$

2.2.2 Game Theoretic Controller with Pareto Strategy

For a 2-player game, the Pareto game or cooperative differential game is formulated by allowing joint strategies to minimize J_c and subject it to the dynamical constraints imposed by Eq. (2.1), such that

$$J_1(u_1^*, u_2^*) \le J_1(u_1, u_2)$$
$$J_2(u_1^*, u_2^*) \le J_2(u_1, u_2) \tag{2.10}$$

where at least one of the inequalities is strict [18]. Consider the infinite time horizon case, the performance indices are defined as

$$J_1(u_1, u_2) = \int_0^\infty (x^TQ_1x + u_1^TR_{11}u_1 + u_2^TR_{12}u_2)\, dt$$

$$J_2(u_1, u_2) = \int_0^\infty (x^TQ_2x + u_1^TR_{21}u_1 + u_2^TR_{22}u_2)\, dt \tag{2.11}$$

where Q_1, Q_2, R_{12}, and R_{21} are positive semi-definite matrices and R_{11}, R_{22} are strictly positive definite matrices. For a convex problem [11], the Pareto solution can be obtained from

$$J_c = \mu_1J_1 + \mu_2J_2 = \int_0^\infty (x^TQ_cx + u^TR_cu)\, dt \tag{2.12}$$

where

$$u^T = [u_1^T \ \ u_2^T] \tag{2.13}$$

$$Q_c = \mu_1 Q_1 + \mu_2 Q_2 \tag{2.14}$$

$$R_c = \begin{bmatrix} \mu_1 R_{11} + \mu_2 R_{21} & 0 \\ 0 & \mu_1 R_{12} + \mu_2 R_{22} \end{bmatrix} \tag{2.15}$$

and the scalar weight coefficients μ_1 and μ_2 satisfy

$$\mu_1 + \mu_2 = 1 \tag{2.16}$$

where $\mu_1 > 0$ and $\mu_2 > 0$.

As shown in Fig. 2.1, the Pareto solution is not unique; different values for μ_i lead to different solutions. One way to determine the weight coefficients is based on the importance between two performance indices [6]. If J_1 is more important than J_2, then μ_1 is assigned a larger value (i.e., $\mu_1 > 0.5$) than μ_2, and vice versa. If both performance indices are equally important, then both weighting coefficients are assigned values of 0.5. After determining the weighting coefficients, the classical optimal control theory can be directly applied to find a minimum feedback solution.

However, it is worthwhile to reiterate that the relative importance of these performance indices is not generally known, until the system's best capabilities are achieved and the trade-offs between the performance indices reached. This is the major drawback for the application of classical optimal control theory to linear regulator problems, since those weighting coefficients are not easily determined such that the best system performance can be achieved.

2.2.3 Mini-Max Strategy

The Mini-Max game or noncooperative zero-sum differential game, is formulated by allowing the minimizer (maximizer) to minimize (maximize) a performance index (J) first, then the maximizer (minimizer) to maximize (minimize) the same performance index (J) subject to the dynamical constraints imposed by Eq. (2.1) and the optimal strategy of

minimizer (maximizer). Considering the infinite time horizon case, we define a performance index as

$$J(u_1, u_2) = \int_0^{\infty} (x^T Q x + u_1^T R_{11} u_1 - u_2^T R_{22} u_2)\, dt \tag{2.17}$$

where Q is a positive semi-definite matrix, and R_{11}, R_{22} are strictly positive definite matrices. When constructing the above performance index, we have assumed that u_1 is the minimizer and u_2 is the maximizer.

The Mini-Max control actions u_1^* and u_2^* are extremals of Eq. (2.17), subjects of the dynamical constraint of Eq. (2.1), if they satisfy following conditions:

$$J(u_1^*, u_2) \leq J(u_1^*, u_2^*) \leq J(u_1^*, u_2) \tag{2.18}$$

The conditions Eq. (2.18) are known as the Mini-Max equilibrium conditions. It has been shown that for these conditions to exist [17], the impulse response functions, $\partial H/\partial u_i$, and the influence functions, λ, must satisfy the following set of Euler-Lagrange equations

$$\left[\frac{\partial H}{\partial u_i}\right]_{u_2^*,\, u_1^*} = 0$$

$$\dot{\lambda}^T = -\frac{\partial H}{\partial x}, \quad \lambda(t_0) = 0 \tag{2.19}$$

and H is the Hamiltonian associated with the performance index J that is given by

$$H = x^T Q x + u_1^T R_{11} u_1 - u_2^T R_{22} u_2 + \lambda^T (Ax + B_1 u_1 + B_2 u_2) \tag{2.20}$$

If linear state feedback is assumed, the control strategies u_1^* and u_2^* satisfying Eqs. (2.1) and (2.18) are

$$u_1^* = -R_{11}^{-1} B_1^T \lambda \tag{2.21}$$

$$u_2^* = +R_{22}^{-1} B_2^T \lambda \tag{2.22}$$

Now, assuming maximizer (u_2) moves first, the optimal control problem becomes that of determining the minimum of the performance index

$$J(u_1, u_2) = \int_0^\infty (x^TQx + u_1^TR_{11}u_1 - u_2^TR_{22}u_2)\ dt \tag{2.23}$$

subject to the dynamic constraints

$$\dot{x} = Ax(t) + B_1u_1(t) + B_2u_2(t)$$

$$u_2^* = R_{22}^{-1}B_2^T\lambda$$

$$y(t) = Cx(t) \tag{2.24}$$

where the second constraint comes from maximizer's strategy. The feedback solutions of the one-side optimal control problem are

$$u_1^* = -\ R_{11}^{-1}B_1^TPx \tag{2.25}$$

$$u_2^* = +\ R_{22}^{-1}B_2^TPx \tag{2.26}$$

where P is the solution of the algebraic Riccati equation

$$-\ PA - A^TP - Q + PB_1R_{11}^{-1}B_1^TP - PB_2R_{22}^{-1}B_2^TP = 0 \tag{2.27}$$

Relationship to H^∞ Optimal Control Problems

A differential game with a Mini-Max strategy for optimal disturbance attenuation can be equivalent to a H^∞ optimal control problem, if one set of controller is a square integratable disturbance that depends on current state [29]. For such a case, the dynamical system is

$$\dot{x} = Ax + B_1u + B_2w, \quad x(0) = 0 \tag{2.28}$$

For the infinite time horizon case, a linear quadratic cost function can be defined as

$$\hat{J} = \int_0^\infty (x^TQx + u^TR_{11}u)\ dt \tag{2.29}$$

where $Q \geq 0,\ R > 0$. Let T_u be the mapping from w to $z \equiv (x, u)$. The disturbance attenuation problem is defined as follows. Find a state feedback controller to minimize $\ll T_u \gg$

$$\inf_u \ll T_u \gg \equiv \gamma^* > 0 \tag{2.30}$$

where

$$\ll T_u \gg \equiv \sup_w \| T_u \| / \| w \|$$

Notice that $\| T_u \|^2 \equiv \hat{J}$ and $\| w \|^2 \equiv \int_0^\infty w^T R_{22} w \, dt$. The optimal state feedback solution and maximum disturbance consists of the saddle point of the game with kernel

$$J_\gamma \equiv \| T_u \|^2 - \gamma^2 \| w \|^2 \equiv \hat{J} - \gamma^2 \| w \|^2, \tag{2.31}$$

where the saddle point (u^*, w^*) statisfies the following inequities

$$J_\gamma(u^*, w) \leq J_\gamma(u^*, w^*) \leq J_\gamma(u, w^*) \tag{2.32}$$

The optimal state feedback control is

$$u^* = -R_{11}^{-1} B_1^T P x \tag{2.33}$$

and its associated maximum disturbance is

$$w^* = \frac{1}{\gamma^2} R_{22}^{-1} B_2^T P x \tag{2.34}$$

Matrix P in Eqs. (2.33) and (2.34) is the smallest nonnegative definite solution of the following equation

$$A^T P + PA + Q - P(B_1 R_{11}^{-1} B_1^T - \frac{1}{\gamma^2} B_2 R_{22} B_2^T) P = 0 \tag{2.35}$$

2.2.4 Stackelberg Strategy

In a 2-player game, if one player has to announce his strategy first due to some bias in the information situation (i.e., this player does not have his opponent's information; however, his opponent knows of his performance index), this player should choose a

Stackelberg solution for himself [25]. A Stackelberg solution for a 2-player nonzero-sum differential game with biased information patterns is stated as follow. Given a 2-player game, Player 1 and Player 2 want to minimize their performance indices

$$J_1(u_1, u_2) = \int_0^{\infty} (x^T Q_1 x + u_1^T R_{11} u_1 + u_2^T R_{12} u_2)\; dt$$

$$J_2(u_1, u_2) = \int_0^{\infty} (x^T Q_2 x + u_1^T R_{21} u_1 + u_2^T R_{22} u_2)\; dt \tag{2.36}$$

respectively. The strategy set (u_1^*, u_2^*) is called a Stackelberg strategy with Player 2 as leader and Player 1 as follower if

$$J_2(u_1^*, u_2^*) \leq J_2(u_1^o(u_2), u_2) \tag{2.37}$$

where

$$J_1(u_1^o(u_2), u_2) = \min_{u_1} J_1(u_1, u_2) \tag{2.38}$$

$$u_1^* = u_1^o(u_2^*) \tag{2.39}$$

As already mentioned in Section 2.1, in a Stackelberg game, the leader (Player 2 in this case) will announce his strategy according to the follower's reaction strategy. Hence, the follower's performance index is the first objective to be minimized in order to determine the leader's strategy. The impulse response functions, $\partial H_1 / \partial u_1$, and the influence functions, λ_1, for Player 1 must satisfy the following set of Euler-Lagrange equations

$$\left[\frac{\partial H_1}{\partial u_1}\right]_{u_1^*} = 0 \tag{2.40}$$

$$\dot{\lambda}_1^T = -\frac{\partial H_1}{\partial x} = -Q_1 x - A^T \lambda_1 \tag{2.41}$$

where H_1 is the Hamiltonian associated with performance index J_1 that is given by

$$H_1 = \frac{1}{2}(x^T Q_1 x + u_1^T R_{11} u_1 + u_2^T R_{12} u_2) + \lambda_1^T (Ax + B_1 u_1 + B_2 u_2) \quad (2.42)$$

Equations (2.40) and (2.41) are the follower's reaction strategy which the leader must choose. Hence, in the next optimization step, Eqs. (2.40) and (2.41) can be treated as additional constrains imposed to the second optimization problem, and should be included in Hamiltonian H_2. The impulse response functions, $\partial H_2 / \partial u_2$, and the influence functions, λ_2, for Player 2, must satisfy

$$\left[\frac{\partial H_2}{\partial u_2}\right]_{u_2^*} = 0 \quad (2.43)$$

$$\dot{\lambda}_2^T = -\frac{\partial H_2}{\partial x} \quad (2.44)$$

$$\dot{\alpha}_2^T = -\frac{\partial H_2}{\partial \lambda_1} \quad (2.45)$$

The Hamiltonian associated with performance index J_2 is

$$H_2 = \frac{1}{2}(x^T Q_2 x + u_1^T R_{21} u_1 + u_2^T R_{22} u_2) + \lambda_2^T (Ax + B_1 u_1 + B_2 u_2) - \alpha_2^T (Q_1 x + A^T \lambda_1) \quad (2.46)$$

The open-loop solutions are given by

$$u_1^* = -R_{11}^{-1} B_1^T P_1 x \quad (2.47)$$

$$u_2^* = -R_{22}^{-1} B_2^T P_2 x \quad (2.48)$$

where P_1 and P_2 are solution of the following coupled equations

$$-P_1 A - A^T P_1 - Q_1 + P_1 B_1 R_{11}^{-1} B_1^T P_1 + P_1 B_2 R_{22}^{-1} B_2^T P_2 = 0 \quad (2.49)$$

$$-P_2 A - A^T P_2 - Q_2 + P_2 B_1 R_{11}^{-1} B_1^T P_1 + P_2 B_2 R_{22}^{-1} B_2^T P_2 + Q_1 P = 0 \quad (2.50)$$

with P being the solution of the following equation

$$-PA - A^T P + PB_1 R_{11}^{-1} B_1^T P_1 + PB_2 R_{22}^{-1} B_2^T P_2 - B_2 R_{11}^{-1} R_{21} R_{11}^{-1} B_1^T P_1 + B_1 R_{11}^{-1} B_1^T P_2 = 0 \quad (2.51)$$

Equations (2.49) and (2.50) are identical to the open-loop Nash strategic solution, except for the additional term Q_1P contained in Eq. (2.50). The extra term reflects that the leader (Player 2) minimizes his cost on the reaction strategy of the follower (Player 1).

2.3 Summary

In this chapter, historical reviews for both the multiobjective optimization problems and game theory are given. The differential game, which is the study of the optimal reactions among all the player under all possible cooperation and/or conflicting situations involved in the dynamical game, is also reviewed. For the noncooperative nonzero-sum differential game, information is exchanged explicitly among all the controllers by utilization of the Nash strategy. This explicit information exchange pattern differs from the classical optimal controller, whose information exchange among all the state feedback controller is through its single performance index. For the multiobjective linear quadratic problem, the Nash strategy is the only controllers that can explicitly exchange dynamic information; for any other strategy (i.e., Pareto strategy, Stackelberg strategy and Mini-Max strategy), controllers can only exchange information implicitly. This is the reason why the noncooperative nonzero-sum differential game is uniquely suited for multiobjective linear quadratic problems involving conflicting situations among players (controllers). Thus, it motivates this research to design game theoretic controllers using Nash strategies.

CHAPTER 3
A ROBUST HOMOTOPY ALGORITHM FOR THE ALGEBRAIC RICCATI EQUATION

3.1 Introduction

State feedback controllers have been successfully developed and used to alter the system response of multiple input/multiple output linear dynamical systems. Optimal state feedback inputs can be generated by utilizing the Linear Quadratic Regulation (LQR) technique [31,32]. The time varying state feedback gain is determined via the solution of a differential Riccati equation. In the infinite time horizon case, the state feedback system is stable if the constant solution of the algebraic Riccati equation is positive definite [17]. Solution methodologies have been intensively investigated in the past two decades [33,34,35]. However, these solution techniques can only be applied to the "standard" algebraic Riccati equation. For example, the "modified" version of the algebraic Riccati equation (i.e., resulted from the system's uncertainty) [36] or the "coupled" algebraic Riccati equation [1,6] (or see Chapters 2 and 4 for detail) cannot be directly solved by any conventional algebraic Riccati equation solver. The "modified" version of the algebraic Riccati equation has the form

$$A^TP + PA - PSP + Q + F(P) = 0$$

In optimal control of reduced large-dimensional systems [37],

$$F(P) = (I - \tau)PSP(I - \tau^T)$$

with τ as the non-symmetric idempotent. In the frequency-dependent uncertainty problem [38],

$$F(P) = \sum_{i=1}^{k} A_iPA_i$$

with $A_i = block\ diag\ (0, \begin{Bmatrix} 0 & \omega \\ -\omega & 0 \end{Bmatrix}, 0)$. In the decentralized H^∞ optimal control problem [39],

$$F(P) = (P - \sum_{i=1}^{k} \tau_i P \tau_i) \sum Q (P - \sum_{i=1}^{k} \tau_i P \tau_i)$$

with $\sum_{i=1}^{k} \tau_i = I_n$.

Two categories of numerical methods have been used to solve algebraic Riccati-type equations, namely, iterative methods and integrative methods. In general, an iterative algorithm takes less computational operating numbers than an integrative method. However, for solving the "modified" or "coupled" algebraic Riccati equation, iterative schemes may diverge due to bad initial guesses. For these types of Riccati equations, an integrative algorithm based on homotopy theory (even thought it may be less computationally efficient) are the more robust solvers. Thus a homotopy algorithm is proposed to solve the algebraic Riccati equation in this chapter. The algorithm developed in this chapter can be extended to solve the "modified" or "coupled" algebraic Riccati equations. The extension to coupled algebraic Riccati equations is presented in Chapter 4.

Homotopy methods have been proven and implemented as globally convergent algorithms to numerically determine the solutions of a system of polynomial equations [8]. Watson [10] developed some generic codes, based on the homotopy theory, which can locate all the zeros of a system of polynomial equations. All the "modified" or "coupled" algebraic Riccati-type equations can be solved by homotopy methods, since they all can be expressed in the form of a system of polynomial equations. However, too much computational effort is required to determine all the solution of the "matrix" algebraic Riccati equation. For an n-by-n system, there exists $n^2/2$ possible solutions for the algebraic Riccati equation, even though the unique positive definite solution is the only solution of interest for the class of problems being examined. Jamshidi and Böttiger [40] first proposed both Q-imbedding and

S-imbedding algorithms to find a solution of the algebraic Riccati equation. Wang and Pan [9] generated a homotopy algorithm to find a solution of the algebraic Riccati equation. Richter, Hodel and Pruett [36] used a homotopy method for the numerical solution of the modified algebraic Riccati equation. Collins, Lawrence and Richter [41] proposed a homotopy algorithm, which involves solving two Riccati equations coupled to two Lyapunov equations, for maximum entropy design. However, solutions for the algebraic Riccati-type equation obtained from either of the above algorithms may not be the positive definite solution, which is the only solution of interests in the optimal control problem.

In this chapter, a "robust" homotopy algorithm which determines the "positive definite" solution of an algebraic Riccati equation is proposed [42]. The algorithm is robust in the sense that the numerical integration will never bifurcate and the resulting solution is guaranteed to be positive definite. The remainder of this chapter is organized as follows. In Section 3.2, a preliminary introduction to homotopy theory is given. In Section 3.3, a robust homotopy algorithm for solving algebraic Riccati equations is presented. Several numerical examples and the summary are provided in Sections 3.4 and 3.5, respectively.

3.2 Preliminary Introduction to the Homotopy Theory

The homotopy theory introduced in this section is mainly based on the research work of Zangwill and Garcia [8]. Given a system of polynomial equations (linear or non-linear) to be solved, the homotopy algorithm starts with a simple solution (a solution to a similar but easily solved system of polynomial equations) and through a series of integrations, reaches the exact solution of the original system of polynomial equations.

Mathematical Model:

Let $\mathfrak{R}^n$ be an Euclidean n space. Given $f(x) : \mathfrak{R}^n \rightarrow \mathfrak{R}^n$ (i.e., $x \in \mathfrak{R}^n$), find the solution, $x^* = [x_1^*, \ \dots \ , \ x_n^*]^T$, of

$$f(x) = 0 \tag{3.1}$$

3.2.1 Homotopy Scheme

Given a system of polynomial equations (Eq. (3.1)) to be solved, the homotopy scheme proceeds as follow:

(1). Identify an easily solved system of polynomial equations, say

$$e(x) = 0 \tag{3.2}$$

where $e(x) : \Re^n \rightarrow \Re^n$. Determine the solution x^0 of Eq. (3.2).

(2). Define a homotopy function $h(x, t)$, such that

$$h(x, 0) = e(x) \tag{3.3}$$

$$h(x, 1) = f(x) \tag{3.4}$$

(3). Generate a path (from $t = 0$ to $t = 1$) which leads the solution from x^0 to x^*.

3.2.2 Mathematical Models of Homotopy Functions, $h(x, t)$

Although there exist numerous forms of homotopy functions, the following are the most common three:

(i). Newton homotopy

$$h(x, t) = f(x) - (1 - t)f(x^0) \tag{3.5}$$

(ii). Fix-point homotopy

$$h(x, t) = (1 - t)(x - x^0) + tf(x) \tag{3.6}$$

(iii). Linear homotopy

$$\begin{aligned} h(x, t) &= tf(x) + (1 - t)e(x) \\ &= e(x) + t(f(x) - e(x)) = 0 \end{aligned} \tag{3.7}$$

Homotopy paths start from any arbitrary point, x^0, for either the Newton homotopy function or the fix-point homotopy function; hence, they are the most popular. However, if one can find an easy solved system of polynomial equations, and it is very “close” to the original system of polynomial equations, a linear homotopy function might be a better

mathematical model to work with. For the latter case, intuitively, one can take larger integration steps, thus requiring less computational effort. It should be noted that all the above homotopy expressions are similar. For examples, $f(x) - f(x^0)$ in Eq. (3.5) and $x - x^0$ in Eq. (3.6) are equivalent to $e(x)$ in Eq. (3.7). The following derivations will be based on the linear homotopy model only, since we can find an easily solved system of polynomial equations (i.e., a Lyapunov equation), and it is very close to the Riccati-type equations.

3.2.3 Path Following Algorithm and Homotopy Differential Equation

The differentiation of any homotopy function with respect to the homotopy variable, t, is zero, since a homotopy function is defined to be zero in (x, t), i.e.,

$$\frac{dh}{dt} = \frac{\partial h}{\partial t} + \frac{\partial h}{\partial x}\frac{\partial x}{\partial t} = 0, \tag{3.8}$$

which yields

$$\dot{x} = -\left(\frac{\partial h}{\partial x}\right)^{-1}\frac{\partial h}{\partial t}. \tag{3.9}$$

Equation (3.9) is referred to as the "homotopy differential equation" or "Davidenko differential equation" [8]. Integration of Eq. (3.9) over the "homotopy variable" t, from 0 to 1, will give a solution for the system of polynomial equations (Eq. (3.1)). Let $x \in D \subset \Re^n$, and $T = \{t \mid 0 \leq t \leq 1\}$. In order to guarantee the path existence, the regularity of H or the inversion of the Jacobian matrix, $\frac{\partial h}{\partial x}$, in Eq. (3.9) should always hold for all (x, t) in H^{-1}, where H^{-1} is defined as the set of all the solutions of $h(x, t)$, i.e.,

$$H^{-1} = \{(x, t) \in D \times T \mid h(x, t) = 0\}. \tag{3.10}$$

The existence of homotopy path which results in the desired solution of a system of polynomial equations is guaranteed by Sard's Theorem (i.e., the homotopy algorithm is globally convergent with "probability one" [10]). Thus, the homotopy algorithm can only fail for points which are in a set of "measure zero." Notice that a set has measure zero in $\Re^n$

if it has no volume in $\Re^n$. For example, a line segment has a positive measure in $\Re^1$, but not in $\Re^n$ for $n \geq 2$, hence a line segment has measure zero in $\Re^n$ for $n \geq 2$.

Theorem 3.2.3 Sard's Theorem (Zangwill [8], pp. 422)

Let $h : \mathcal{H} \rightarrow \subset \Re^q \rightarrow \Re^n$, where $\mathcal{H}$ is the closure of an open set and h be $\mathcal{C}^k$. If $k \geq 1 + \max\{0, q - n\}$, then h is regular for almost all ε, where

$$h(\cdot) = \varepsilon$$

❑

Corollary 3.2.3 Extended Sard's Theorem (Zangwill [8], pp. 422)

Let $D \subset \Re^n$ be the closure of an open set, $h : D \times T \rightarrow \Re^n$ be $\mathcal{C}^2$, and $f : D \rightarrow \Re^n$ be $\mathcal{C}^1$, then the following three statements are equivalent,

(i) h will be regular for almost all ε,

(ii) h will be regular at $\bar{t}$ for almost all ε,

(iii) f will be regular for almost all ε. ❑

For a homotopy function, $q = n + 1$, if $k \geq 2$, then $h(x, t)$ is regular for almost all ε. In other words, as long as $h(x, t) \subset \mathcal{C}^2$, Sard's theorem states that for an arbitrary e, $H(x, t)$ is almost assuredly regular. It can be shown that all the systems of polynomial equations are continuous and hence all algebraic Riccati equations are continuous. Thus, homotopy functions for the algebraic Riccati equations belong to $\mathcal{C}^2$. It infers that even if the integration path bifurcates (or, the Jacobian matrix is singular) in a homotopy algorithm, we can slightly perturb the stating point of the homotopy equation such that a new integration path is guaranteed to exist. Sard's theorem has been utilized by most current homotopy algorithms [10,36,41] to bypass the bifurcation points along the homotopy path.

3.3 Homotopy Algorithm to Solve an Algebraic Riccati Equation

An algebraic Riccati equation is defined as

$$PA + A^TP - PBR^{-1}B^TP + Q = 0 \tag{3.11}$$

where $A,\ P,\ Q \in \Re^{n\times n},\ B \in \Re^{n\times m}$ and $R \in \Re^{m\times m}$. Matrices P and R are positive definite, Q is positive semi-definite. Before developing the homotopy algorithm which solves an algebraic Riccati equation (Eq. (3.11)), we digress for a moment to discuss the vector operators and Kronecker products. These properties will be used to "vectorize" the matrix equation which results from the homotopy process.

Prop. 3.3.1 Vector operators and properties of Kronecker products (Graham [43])

Given matrices $A \in \Re^{m\times n}$, $B \in \Re^{r\times s}$, $C \in \Re^{n\times n}$, $D \in \Re^{m\times m}$ and $Y \in \Re^{n\times r}$, the following vector operators properties exist.

(i). Kronecker product:

$$A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & \cdots & a_{1n}B \\ a_{21}B & a_{22}B & \cdots & a_{2n}B \\ \vdots & \vdots & & \vdots \\ a_{m1}B & a_{m2}B & \cdots & a_{mn}B \end{bmatrix} \in \Re^{mr\times ns}$$

(ii). Kronecker sum:

$$C \oplus D = C \otimes I_m + I_n \otimes D$$

where I_m and I_n are $m \times m$ and $n \times n$ the identity matrices, respectively.

(iii). Vector of a matrix:

$$Vec(A) = [a_{11}\ a_{21}\ \dots\ a_{m1}|a_{12}\ a_{22}\ \dots\ a_{m2}|\dots\ \dots\ \dots\ \dots|a_{1n}\ a_{2n}\ \cdots\ a_{mn}]^T$$

(iv). Vector of a matrix sum (distributive property):

$$Vec(A + B) = Vec(A) + Vec(B)$$

(v). Vector of a matrix product:

$$Vec(AYB) = (B^T \otimes A)VecY$$

❑

Returning to the development of a homotopy algorithm, the following derivation is based on a linear homotopy model, since an easily solved system of polynomial equations can be found (i.e., a Lyapunov equation). Let

$$f(P) = PA + A^TP - PBR^{-1}B^TP + Q = 0 \tag{3.12}$$

and choose a starting point from the non-trivial solution of $e(P)$

$$e(P) = P(A + aI) + (A + aI)^TP - PBR^{-1}B^TP = 0 \tag{3.13}$$

where a is either zero or a positive random number plus the negation of smallest non-positive eigenvalues of the matrix A (if A has any); the quantity a is chosen to ensure that $(A + aI)$ is strictly positive definite. Notice that similar techniques of shifting eigenvalues of original system matrix have been found in Ref. 9 which uses a "different" homotopy function to solve an algebraic Riccati equation and in Ref. 41 which solves the modified algebraic Riccati equations resulting from the maximum entropy problems. The reason for matrix $(A + aI)$ to be positive definite will be discussed in Section 3.3.1. Based on Eqs. (3.13) and (3.12), the matrix form of the homotopy function can be rewritten as

$$\begin{aligned} h(P,t) &= e(P) + t\ (f(P,t) - e(P)) \\ &= P(A + (1 - t)aI) + (A + (1 - t)aI)^TP - PBR^{-1}B^TP + tQ = 0 \end{aligned} \tag{3.14}$$

Differentiating Eq. (3.14) with respect to the "homotopy variable" t yields

$$\dot{P}A_s + A_s^T\dot{P} = -Q + 2aP \tag{3.15}$$

where

$$A_s = A + (1 - t)aI - BR^{-1}B^TP$$

$\dot{P}$ in Eq. (3.15) can be solved by standard Lyapunov solvers. However, in order to pave the way for future use (i.e., the solution of coupled Lyapunov equations in Chapter 4), another solution algorithm utilizing the properties of Kronecker products is developed as follow. Notice that the propose algorithm utilizes the symmetric property of $\dot{P}$ in conjunction with the Kronecker sum to improve its efficiency.

Using Prop. 3.3.1, Eq. (3.15) can be vectorized as

$$(A_s^T \oplus A_s^T)Vec(\dot{P}) = -Vec(Q) + 2aVec(P) \tag{3.16}$$

where

$$Vec(\dot{P}) = [\dot{p}_{11}\ \dot{p}_{21}\ \cdots\ \dot{p}_{n1}|\dot{p}_{12}\ \dot{p}_{22}\ \cdots\ \dot{p}_{n2}|\cdots\ \cdots\ \cdots\ \cdots|\dot{p}_{1n}\ \dot{p}_{2n}\ \cdots\ \dot{p}_{nn}]^T$$

$$Vec(P) = [p_{11}\ p_{21}\ \cdots\ p_{n1}|p_{12}\ p_{22}\ \cdots\ p_{n2}|\cdots\ \cdots\ \cdots\ \cdots|p_{1n}\ p_{2n}\ \cdots\ p_{nn}]^T$$

$$Vec(Q) = [q_{11}\ q_{21}\ \cdots\ q_{n1}|q_{12}\ q_{22}\ \cdots\ q_{n2}|\cdots\ \cdots\ \cdots\ \cdots|q_{1n}\ q_{2n}\ \cdots\ q_{nn}]^T$$

Comparing Eqs. (3.9) and (3.16), one observes that the Kronecker sum,

$$A_s^T \oplus A_s^T$$

defined in Eq. (3.16) is actually the Jacobian matrix.

The proposed homotopy scheme thus far involves the solution of an n^2 system of polynomial equations at each integration step for $\dot{P}$. By utilizing the symmetric property of the Riccati matrix in conjunction with the Kronecker sum, the number of equations to be solved can be reduced to $\frac{n(n+1)}{2}$.

Once a current value of $\dot{P}_k\ (=\dot{P}(t_k))$ is available, the new solution is simply

$$P_{k+1} = P_k + \dot{P}_k h_k \tag{3.17}$$

where "h_k" is the current integration step size. After obtaining P_{k+1}, let $k = k + 1$ and repeat the process to find a new P_{k+1} until $t_k = 1$. Then, solving an algebraic Riccati Equation is equivalent to following the homotopy path equation (Eq. (3.14)) (from $t = 0$ to $t = 1$) with a "positive" starting point, P_0, which is the inverse of the solution of the following Lyapunov equation

$$(A + aI)V + V(A + aI)^T - BR^{-1}B^T = 0 \tag{3.18}$$

Notice that the homotopy differential equation (Eq. (3.16)) is in the form of a system of linear uncoupled first order ordinary differential equations for $\dot{P}$. Several numerical methods (e.g., Euler method, Runge-Kutta methods, predictor-corrector methods, etc.) can be utilized. Numerical integration using Euler method is not very accurate when the

integration step, h_k, is too large (i.e., error at each integration step is $O(h_k^2)$). Hence, some of the existing homotopy algorithms utilize a correction formula (e.g., Newton's formula) to reduce the integration error [36,41]. Instead of using those predictor-corrector methods, the proposed homotopy algorithm uses a variable step size Runge-Kutta code (i.e., ode45.m in MATLAB) to implement the numerical integration. Numerical results show the solution is fairly accurate (numerical examples are in Section 3.4). It should be noted that in solving the reduced form of Eq. (3.16), the Jacobian matrix is QR factorized and back substituted at each integration step to improve numerical efficiency.

3.3.1 Positive Definite Solution and Path Existence

A detailed proof that the "Jacobian matrices" are non-singular at each integration point on the homotopy path and the final solution ($t = 1$) is positive definite will be presented in this section.

Theorem 3.3.1.1 (Gopal [44], pp. 211)

Given a time-invariant system

$$\dot{x} = Ax + Bu \tag{3.19}$$

where $A \in \Re^{nxn}$, $B \in \Re^{nxm}$, the following statements are equivalent:

1. The system is completely controllable.
2. The rows of $e^{At}B$ are linear independent over the real field for all $t \in [0, \infty)$.
3. The state covarience matrix is positive definite, i.e.,

$$V > 0, \quad V \triangleq \int_0^\infty e^{At} B R^{-1} B^T e^{A^T t}\, dt, \quad R > 0$$

4. rank $C = n, \quad C \triangleq [B,\ AB,\ \cdots,\ A^{n-1}B]$.

❑

Corollary 3.3.1.1 (Skelton [32], pp. 211)

Let $R > 0$. If all eigenvalues of A in Eq. (3.19) are negative definite, then rank $C = n$ iff $V > 0$ where V satisfies

$$AV + VA^T + BR^{-1}B^T = 0$$

❑

Corollary 3.3.1.2

If (A, B) is a controllable pair and the matrix, $A + aI$, is positive definite, then the non-trivial solution, P, from Eq. (3.13) is positive definite.

Proof:

From Statement 2 in Theorem 3.3.1.1, if (A, B) is controllable, then the rows of $e^{At}B$ are linear independent over the real field for all $t \in [0, \infty)$. (Notice that in this proof, the symbol, t, stands for time variable, but not for "homotopy variable.") Based on the matrix multiplication commutativity between matrices At and atI,

$$e^{(A+aI)t}B = e^{At}e^{atI}B = e^{at}(e^{At}B)$$

Therefore, $(A + aI, B)$ is controllable, since the rows of $e^{at}(e^{At}B)$ are linear independent over the real field for all $t \in [0, \infty)$. Based on Corollary 3.3.1.1 and $A + aI$ being positive definite, the controllability grammian V is positive definite and satisfies

$$(-A - aI)V + V(-A - aI)^T + BR^{-1}B^T = 0 \tag{3.20}$$

where $R \in \Re^{m \times m}$ is positive definite. Substitution of P^{-1} for V, and pre-multiplying and post-multiplying the above equation by P yields Eq. (3.13). Hence, the solution, P, from Eq. (3.13) is positive definite.

❑

It can be shown that the "starting point" (i.e., solution of Eq. (3.13)) to the proposed integrative scheme is positive definite based on Theorem 3.3.1.1, Corollaries 3.3.1.1 and 3.3.1.2.

Based on Corollary 3.3.1.2, the "shifting parameter," α, is chosen as either zero or a positive random number plus the absolute value of smallest non-positive eigenvalues of the matrix A (if A has one). The reason to add a random number to α is to make the matrix, $A + \alpha I$, strictly positive definite, thus the "starting point" to the homotopy differential equation (Eq. (3.13)) is positive definite.

Notice that, from Corollary 3.3.1.1, if the solution, V, of the Lyapunov equation (Eq. (3.18) or (3.20)) has a non-positive eigenvalue, then the systems, (A, B) and $(A + \alpha I, B)$, are not controllable. For the non-controllable system, the homotopy algorithm will stop at the "starting point" and give no solutions for the algebraic Riccati equation, since the inversion of V is not positive definite.

Up to this point, the starting point, P_0, of the proposed homotopy algorithm is proven to be positive definite. It can also be shown the starting Jacobian matrix is non-singular. The following theorems are need for the proof of the Jacobian matrix being non-singular.

Theorem 3.3.1.2 (Graham [43], pp. 30)

If $\{\alpha_i\}$ and $\{\beta_j\}$ are all the eigenvalues associated with matrices A and B, then $\{\alpha_i + \beta_j\}$ are all the eigenvalues of $A \oplus B$.

❑

Theorem 3.3.1.3 (Wiberg [45], pp. 214)

If P is the positive definite solution of the algebraic Riccati equation, the closed-loop system is asymptotically stable (i.e., the real parts of all eigenvalues of the closed-loop system matrix are negative).

❑

Based on Theorems 3.3.1.2 and 3.3.1.3, the starting Jacobian matrix in Eq. (3.16) is non-singular, since the starting A_s from Eq. (3.15), is negative definite (i.e., $A_s = A + \alpha I - BR^{-1}B^T P_0 < 0$). Thus the integration cannot bifurcate at the starting point, and $\dot{P}_0$ and the next solution P_1 exist

$$P_1 = P_0 + \dot{P}_0 h_0 \tag{3.21}$$

where h_0 is the integration step at point t_0 and $\dot{P}_0$ is the derivative of P with respect to homotopy parameter, t, at point t_0 and satisfies the following equation

$$\dot{P}_0 A_{s_0} + A_{s_0}^T \dot{P}_0 = -Q + 2\alpha P_0 \tag{3.22}$$

where

$$A_{s_0} = A + \alpha I - BR^{-1}B^T P_0.$$

In order to show under certain conditions that P_1 is always positive, we introduce the following theorems.

Theorem 3.3.1.4 (Gantmacher [46], pp. 224)

All the eigenvalues of a real constant matrix, A, have negative real parts, iff the matrix equation

$$A^T P + PA = -E$$

(with any positive definite matrix, E) has a positive definite solution, P.

❑

Theorem 3.3.1.5 (Molinari [47])

Let (A, B) be a controllable pair. If A is non-singular, then there exists a positive semi-definite solution P for the following Lyapunov-type algebraic equation

$$PA + A^T P - PBR^{-1}B^T P = 0$$

such that

$$\mathrm{Re}(s_i) < 0 \text{ and } s_i^2 = \lambda_i^2 \text{ with } i = 1,\ 2,\ \ldots,\ n$$

where λ_i are the open-loop poles and s_i are the closed-loop poles with the state feedback gain matrix $K = R^{-1}B^TP$.

❑

First, we claim that if $\alpha = 0$ in Eq. (3.22), then $\dot{P}_0$ is positive (recall that $\alpha = 0$ when all eigenvalues of the system matrix, A, have positive real parts). Based on Eq. (3.22) and Theorem 3.3.1.4, the proof is obvious. Thus P_1 is positive, since $P_1 = P_0 + \dot{P}_0 h_0$ and $h_0 > 0$. Now, if $\alpha > 0$ (i.e., the system matrix, A, has at least one eigenvalue which is non-positive), let

$$\alpha = \sigma_n + \delta$$

where σ_n is the absolute value of smallest non-positive eigenvalue of A and δ is a positive random number. The following theorem will show that P_1 is positive definite if $h_0 < \frac{\delta}{\alpha}$ for $\alpha > 0$ case.

Theorem 3.3.1.6

Let P_1 be the solution of the homotopy equation (Eq. (3.14)) at $t = t_1$ $(= h_0)$

$$P_1(A + (1 - h_0)\alpha I) + (A + (1 - h_0)\alpha I)^T P_1 - P_1 B R^{-1} B^T P_1 + h_0 Q = 0 \tag{3.23}$$

or

$$P_1 A_{s_1} + A_{s_1}^T P_1 = - P_1 B R^{-1} B^T P_1 - h_0 Q \tag{3.24}$$

where

$$A_{s_1} = A + (1 - h_0)\alpha I - B R^{-1} B^T P_1 \tag{3.25}$$

and $\alpha = \sigma_n + \delta > 0$, where $\sigma_n \geq 0$ is the absolute value of smallest non-positive eigenvalue of A (if A has any). Then P_1 is positive definite if

$$h_0 < \frac{\delta}{\alpha}$$

Proof:

Define A_α as the initial starting open-loop system matrix

$$A_\alpha \equiv A + \alpha I$$

and let $S = BR^{-1}B^T$. The solution from the first integration step is symboled as

$$P_1 \equiv P_0 + \varDelta P_0$$

Substitution of A_α, S and P_1 into Eq. (3.23) yields as

$$(P_0 + \varDelta P_0)(A_\alpha - h_0\alpha I) + (A_\alpha - h_0\alpha I)^T(P_0 + \varDelta P_0) \\ - (P_0 + \varDelta P_0)S(P_0 + \varDelta P_0) + h_0Q = 0 \qquad (3.26)$$

Subtract Eq. (3.20) from Eq. (3.26)

$$- 2h_0\alpha P_0 - 2h_0\alpha\varDelta P_0 + \varDelta P_0A_\alpha + A_\alpha^T\varDelta P_0 \\ - P_0S\varDelta P_0 - \varDelta P_0SP_0 - \varDelta P_0S\varDelta P_0 + h_0Q = 0 \qquad (3.27)$$

and multiply Eq. (3.22) by h_0

$$\varDelta P_0(A_\alpha - SP_0) + (A_\alpha - SP_0)^T\varDelta P_0 = h_0(-Q + 2\alpha P_0) \qquad (3.28)$$

Subtract Eq. (3.28) from (3.27)

$$- 2h_0\alpha\varDelta P_0 = \varDelta P_0S\varDelta P_0 \qquad (3.29)$$

From Eq. (3.28), we can show, based on Theorem 3.3.1.4 and $A_\alpha - SP_0 < 0$, that

(1) $\varDelta P_0 > 0$, if $-Q + 2\alpha P_0 < 0$

(2) $\varDelta P_0 < 0$, if $-Q + 2\alpha P_0 > 0$

In Eq. (3.22), if we can choose a proper "α" such that $Q - 2\alpha P_0 > 0$ or $Q - 2\alpha P_0 < 0$, then the inversion of $\varDelta P_0$ exists. Hence, Eq. (3.29) can be written as

$$- 2h_0\alpha I = \varDelta P_0S.$$

Substitution of "$P_0 + \varDelta P_0$" for P_1, "$-2h_0\alpha I$" for $S\varDelta P_0$, and "$BR^{-1}B^T$" for S in Eq. (3.25) yields

$$A_{s_1} = A_{s_0} + h_0\alpha I. \qquad (3.30)$$

According to Molinari's mirror image property (Theorem 3.3.1.5), the largest eigenvalue of

A_{s_0} is $-\delta$, since the smallest eigenvalue of A_α is δ. Hence if

$$h_0 < \frac{\delta}{\alpha}, \tag{3.31}$$

then A_{s_1} is negative definite. From Eq. (3.24) and Theorem 3.1.1, one can show that P_1 is positive definite.

❑

We have shown that if either $\alpha = 0$ or Eq. (3.31) holds (but not both), then P_1 is positive definite. For a positive definite P_1, one can prove that $\dot{P}_1$ exists from Eq. (3.16) and Theorem 3.3.1.3. Any successive integration solution on the homotopy path described by Eq. (3.14) is guaranteed to exist and to be positive definite, and its associated Jacobian matrix is also guaranteed to be non-singular, based on the following proposed theorem.

Theorem 3.3.1.7

The necessary condition for new solution, P_{k+1}, in Eq. (3.14) to be positive definite and the new Jacobian matrix, $A^T_{s_{k+1}} \oplus A^T_{s_{k+1}}$, to be non-singular, where

$$P_{k+1} = P_k + \dot{P}_k h_k$$

$$A_{s_{k+1}} = A + (1 - t_{k+1})\alpha I - BR^{-1}B^T P_{k+1}$$

is that current integration step size, h_k, is less than current integration point, t_k, where $k \geq 1$.

Proof:

We need to consider two possibilities: $\alpha = 0$ and $\alpha > 0$. For $\alpha = 0$, P_{k+1} is positive definite, since $\dot{P}_k$ (from Eq. (3.15)) and P_k are positive definite, and h_k is positive. The Jacobian matrix, $A^T_{s_{k+1}} \oplus A^T_{s_{k+1}}$, is non-singular (more precisely negative definite), based on Theorem 3.3.1.4.

For $\alpha > 0$, we first write Eq. (3.14) at t_k

$$P_k(A + (1 - t_k)aI) + (A + (1 - t_k)aI)^T P_k$$
$$- P_k BR^{-1}B^T P_k + t_k Q = 0 \qquad (3.32)$$

and Eq. (3.15) at t_k

$$\dot{P}_k A_{s_k} + A_{s_k}^T \dot{P}_k = - Q + 2aP_k \qquad (3.33)$$

where

$$A_{s_k} = A + (1 - t_k)aI - BR^{-1}B^T P_k$$

Then, multiply Eq. (3.22) by h_k and add the result to Eq. (3.32)

$$P_{k+1}A_{s_k} + A_{s_k}^T P_{k+1} = - P_k BR^{-1}B^T P_k - t_k Q + h_k(- Q + 2aP_k) \qquad (3.34)$$

Note that from Eq. (3.32), for $t_k \neq 0$ (since $k \geq 1$),

$$- Q + 2aP_k = \frac{1}{t_k}[P_k(A + aI - BR^{-1}B^T P_k) + (A + aI - BR^{-1}B^T P_k)^T P_k +$$
$$P_k BR^{-1}B^T P_k]$$

Substitute the right hand side (R.H.S.) of the above equation for $- Q + 2aP_k$ into Eq. (3.34)

$$P_{k+1}A_{s_k} + A_{s_k}^T P_{k+1} = - (1 - \frac{h_k}{t_k})P_k BR^{-1}B^T P_k - t_k Q$$
$$+ \frac{1}{t_k}[P_k(A + aI - BR^{-1}B^T P_k) +$$
$$(A + aI - BR^{-1}B^T P_k)^T P_k]$$

Notice that the necessary condition for the R.H.S. of the above equation to be negative definite is $h_k < t_k$. In other words, if the current integration step, h_k, is less than current integration point, t_k, then the R.H.S. of the above equation is negative definite. For a proper choice of h_k, (i.e., $h_k < t_k$), P_{k+1} is positive definite, based on Theorem 3.3.1.4, since the R.H.S. of the above equation is negative definite and A_{s_k} is also negative definite. Moreover, P_{k+1} is the "positive" solution of the homotopy equation (Eq. (3.14)) at t_{k+1}, i.e.,

$$P_{k+1}A_{s_{k+1}} + A_{s_{k+1}}^T P_{k+1} = - P_{k+1}BR^{-1}B^T P_{k+1} - t_{k+1}Q \qquad (3.35)$$

where

$$A_{s_{k+1}} = A + (1 - t_{k+1})\alpha - BR^{-1}B^TP_{k+1}$$

It can be further concluded that $A_{s_{k+1}}$ is negative based on Theorem 3.3.1.4, since P_{k+1} is positive definite and the R.H.S. of Eq. (3.35) is negative definite. The Jacobian matrix, $A^T_{s_{k+1}} \oplus A^T_{s_{k+1}}$, is non-singular (more precisely negative definite), based on Theorem 3.3.1.2

❑

3.3.2 A Robust Homotopy Algorithm

The difference between the proposed algorithm and any other generic homotopy algorithm is that the solution is guaranteed to be positive definite. The path following algorithm is summarized as follow:

Step 1. Transform the algebraic Riccati equation (Eq. (3.11)) into the homotopy equation (Eq. (3.14)).

Step 2. Differentiate the homotopy equation (Eq. (3.14)) to get the homotopy differential equation (Eq. (3.15)).

Step 3. Vectorize Eq. (3.15) to get Eq. (3.16).

Step 4. Determine a starting point from the inverse of solution of Eq.(3.18). Notice that if the solution of Eq.(3.18) has any non-positive eigenvalue (i.e., the system is not controllable), then go to step 10 and stop.

Step 5. $t_0 = 0$.

Step 6. QR factorize current reduced ($\frac{n(n+1)}{2} \times \frac{n(n+1)}{2}$, instead of $n^2 \times n^2$) Jacobian matrix, $A^T_{s_k} \oplus A^T_{s_k}$, in Eq. (3.16), and solve for $\dot{P}_k$.

Step 7. Use a modified Runge-Kutta variable step size algorithm to numerically integrate Eq. (3.16). (Note: When new solution is not positive definite, change the integration step size such that (i) current integration step, h_K, is less

than the current homotopy variable, t_k, and (ii) $h_0 < \frac{\delta}{\alpha}$, if the system matrix, A, has any eigenvalue which has a non-positive real part).

Step 8. $t_{k+1} = t_k + h_k$ and $P_{k+1} = P_k + \dot{P}_k h_k$, where h_K is the variable sized integration step generated from "ode45.m."

Step 9. Check if $t_{k+1} < 1$.

If true, let $k = k + 1$, go to step 6, otherwise, go to step 10.

Step 10. Stop.

3.4 Numerical Examples

To demonstrate the robustness of our homotopy algorithm, several examples are given.

Example 3.4.1: A non-controllable system (Wang and Pan [9])

$$A = \begin{bmatrix} -13.276 & 3.884 & -568.251 & -30.369 & 0.000 \\ 12.252 & -6.147 & 522.304 & 28.028 & 0.000 \\ 1.972 & 2.136 & -2.562 & 0.000 & 0.000 \\ 5.698 & 0.000 & 12.545 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.000 & 62.369 \\ -20.034 & -57.560 \\ 24.011 & 0.000 \\ 0.000 & 0.000 \\ 0.000 & 0.000 \end{bmatrix}, \; R = Diag([1 \;\; 1]), \; Q = Diag([1 \;\; 1 \;\; 1 \;\; 1 \;\; 2])$$

The open-loop system matrix A has eigenvalues of $-1.1497 \pm 13.9866i$, -1.0497, -18.6359, and 0. Since matrix A has non-positive eigenvalues, δ and α are chosen to be 0.2191 (a positive random number) and 18.8548, respectively. The systems, (A, B) and $(A + \alpha I, B)$, are not controllable and the rank of their associated controllability matrix $C(A, B)$ is 4. Conventional homotopy algorithms [9,10] cannot automatically check if the system is controllable. Thus, they will generate a "solution" to the algebraic Riccati equation, even though the system is not controllable. The proposed homotopy algorithm detects this non-controllable situation at the first step, since the Lyapunov equation (Eq.

(3.18) or (3.20)) has a non-positive eigenvalue (i.e., the eigenvalues for this Lyapunov solution are 0, 1.0368×10^{-2}, 1.0883×10^{1}, 4.1621×10^{2}, 1.2651×10^{6}).

Example 3.4.2: A controllable open-loop system with positive eigenvalues

$$A = \begin{bmatrix} 2 & 1 & 0 & 1 \\ 1 & 2 & 1 & 0 \\ 0 & 1 & 2 & 1 \\ 1 & 0 & 1 & 3 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$R = Diag([1 \ \ 1]), \ Q = Diag([1 \ \ 1 \ \ 1 \ \ 1])$$

Matrix A has eigenvalues of $[4.3429, \ 2.4707, \ 2.0000, \ 0.1864]$. Since matrix A has no non-positive eigenvalues, both δ and α are chosen to be 0. The pair (A, B) is controllable. The positive definite solution, P, is

$$P = \begin{bmatrix} 10.5367 & 3.7065 & 18.3878 & 37.1000 \\ 3.7065 & 8.3179 & 20.6206 & 17.8166 \\ 18.3878 & 20.6206 & 112.6704 & 135.1987 \\ 37.1000 & 17.8166 & 135.1987 & 224.7066 \end{bmatrix}$$

This "efficient" homotopy scheme takes 0.6179 mega flops (using "flops" command in MATLAB) to reach the solution P. The 2-norm of $f(P)$ is 3.5422×10^{-7}, where $f(P) = PA + A^TP - PBR^{-1}B^TP + Q$. It should note that the other homotopy scheme, which dose not use symmetric property of the Riccati solution, requires 0.7354 mega flops to obtain the solution.

Example 3.4.3: A controllable open-loop system with non-positive eigenvalues

$$A = \begin{bmatrix} -2 & 1 & 0 & 1 \\ 1 & -2 & 1 & 0 \\ 0 & 1 & -2 & 1 \\ 1 & 0 & 1 & -2 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$R = Diag([1 \ \ 1]), \ Q = Diag([1 \ \ 1 \ \ 1 \ \ 1]).$$

Matrix A has eigenvalues of $[0, \ -2, \ -2, \ -4]$. Since matrix A has non-positive eigenvalues, δ and α are chosen to be 0.8977 (a positive random number) and 4.8977, respectively. The systems, (A, B) and $(A + \alpha I, B)$, are controllable. The positive definite solution, P, is

$$P = \begin{bmatrix} 0.4895 & 0.3085 & 0.2785 & 0.3378 \\ 0.3085 & 0.4895 & 0.3378 & 0.2785 \\ 0.2785 & 0.3378 & 0.5561 & 0.3703 \\ 0.3378 & 0.2785 & 0.3703 & 0.5561 \end{bmatrix}$$

The "efficient" homotopy scheme takes 1 . 9454 mega flops to reach the solution P. The 2-norm of $f(P)$ is 4.6824×10^{-6}, where $f(P) = PA + A^TP - PBR^{-1}B^TP + Q$. Notice that the other homotopy scheme, which does not use the symmetric property of the Riccati solution, requires 2 . 2834 mega flops to obtain the solution.

3.5 Summary

In this chapter, a robust homotopy algorithm to which solves the algebraic Riccati equation is presented. A brief introductory for homotopy theory is first given. Next, a proof is provided to show that with proper selections of integration step sizes, the proposed homotopy algorithm is guaranteed to generate the positive definite solution of the algebraic Riccati equation. The proposed homotopy algorithm utilizes the symmetric property of Riccati matrix in conjunction with the Kronecker sum to make itself more efficient. The proposed homotopy is robust in the sense that (1) by proper selection of the integration step size, bifurcation will never occur and (2) the solution is guaranteed to be positive definite provided the system is controllable. To this end, we extend the application of this research to the solution of the "modified" or "coupled" algebraic Riccati equations in Chapter 4.

CHAPTER 4
AN EFFICIENT HOMOTOPY ALGORITHM FOR THE COUPLED ALGEBRAIC RICCATI EQUATIONS

4.1 Introduction

As mentioned in Chapter 2, it has been found that many multiobjective dynamical systems have conflict of interest between controller subsystems [1,6]. A noncooperative differential game was proposed by Issaacs [3] to handle situations where conflict of interest exists. The first research work which constructed game theoretic feedback controllers for a linear quadratic regulator problem is attributed to Starr and Ho [4]. In comparison with all the other game theoretic strategies (i.e., Pareto, Mini-max and Stackelberg), a Nash strategy is the only strategy in which the state feedback controllers can exchange dynamic information explicitly for multiobjective linear quadratic regulator problems. Thus, a Nash strategy is uniquely suited for multiobjective linear quadratic problems involving conflict among controllers. Furthermore, for the infinite time horizon problems, it has been shown that the game theoretic state feedback gain matrices are constant in time and are determined from solutions of coupled algebraic Riccati equations [1,5]. As a clarification, throughout this dissertation, the author uses the phrase "game theoretic state feedback controller" as the abbreviation of "a game theoretic state feedback controller using a Nash strategy."

Numerous authors have contributed to solutions of the coupled algebraic Riccati equations. For examples, an iterative scheme based on Newton's method was developed by Krikelis and Rekasius [5] to solve zero-sum differential games. Another algorithm, which is based on a conjugate gradient technique to solve cooperative differential games was formulated by Innocenti and Schmidt [7]. Strikant and Basar [48] proposed an iterative scheme to solve noncooperative nonzero-sum differential games with "weakly" coupled

players. Later on, a modified Krikelis's scheme which utilizes the properties of Kronecker products to solve 2-player nonzero-sum noncooperative differential games with "strongly" coupled players was proposed by Fitz-Coy and Jang [1]. However, convergency of all these iterative schemes greatly depend upon a good initial guess. A bad initial guess may result in these iterative schemes failing to converge.

Homotopy methods have been proven and implemented as globally convergent algorithms to numerically determine the solutions of a system polynomial equation [8]. Homotopy methods can also be utilized to solve the algebraic Riccati-type equations, since these equations can be converted into the form of systems of polynomial equations. In Chapter 3, we developed a robust homotopy algorithm for the "classical" algebraic Riccati equation. Richter, Hodel and Pruett [36] used a homotopy method for the numerical solution of the modified algebraic Riccati equation. Collins, Lawrence and Richter [41] proposed a homotopy algorithm, which is capable of solving two Riccati equations weakly coupled to two Lyapunov equations. In this chapter, a homotopy algorithm is proposed to solve strongly coupled algebraic Riccati equation which results from solving multiobjective linear quadratic regulators problems using the Nash strategy. The proposed algorithm guarantees to give a solution of coupled algebraic Riccati equations. The proposed homotopy algorithm is also very efficient since it utilizes the symmetric property of coupled algebraic Riccati equations in conjunction with the Kronecker sum.

This chapter is organized in the following manner. In Section 4.2, an efficient (as opposed to generic homotopy codes) "linear" homotopy algorithm for the solution of coupled algebraic Riccati equations is presented. Numerical examples and a summary are provided in Sections 4.3 and 4.4, respectively.

4.2 Homotopy Algorithm to Solve Coupled Algebraic Riccati Equations

A "strongly" coupled algebraic Riccati equations is defined as (as opposed to the "weakly" coupled algebraic Riccati equations [48])

$$P_1A + A^TP_1 + Q_1 - P_1B_1R_{11}^{-1}B_1^TP_1 - P_1B_2R_{22}^{-1}B_2^TP_2$$
$$- P_2B_2R_{22}^{-1}B_2^TP_1 + P_2B_2R_{22}^{-1}R_{12}R_{22}^{-1}B_2^TP_2 = 0 \tag{4.1}$$

$$P_2A + A^TP_2 + Q_2 - P_2B_2R_{22}^{-1}B_2^TP_2 - P_2B_1R_{11}^{-1}B_1^TP_1$$
$$- P_1B_1R_{11}^{-1}B_1^TP_2 + P_1B_1R_{11}^{-1}R_{21}R_{11}^{-1}B_1^TP_1 = 0 \tag{4.2}$$

where matrices A, P_j and $Q_j \in \Re^{n\times n}$, $B_j \in \Re^{n\times m_j}$, $R_{j1} \in \Re^{m_1\times m_1}$, $R_{j2} \in \Re^{m_2\times m_2}$ with $j = 1,2$. Matrices P_j and R_{jj} are positive definite, Q_j is positive semi-definite with $j = 1,2$. Let

$$f_1(P_1,P_2) = P_1A + A^TP_1 + Q_1$$
$$- P_1B_1R_{11}^{-1}B_1^TP_1 - P_1B_2R_{22}^{-1}B_2^TP_2$$
$$- P_2B_2R_{22}^{-1}B_2^TP_1 + P_2B_2R_{22}^{-1}R_{12}R_{22}^{-1}B_2^TP_2 = 0 \tag{4.3}$$

$$f_2(P_1,P_2) = P_2A + A^TP_2 + Q_2$$
$$- P_2B_2R_{22}^{-1}B_2^TP_2 - P_2B_1R_{11}^{-1}B_1^TP_1$$
$$- \mathrm{P}_1\mathrm{B}_1\mathrm{R}_{11}^{-1}\mathrm{B}_1^T\mathrm{P}_2 + \mathrm{P}_1\mathrm{B}_1\mathrm{R}_{11}^{-1}\mathrm{R}_{21}\mathrm{R}_{11}^{-1}\mathrm{B}_1^T\mathrm{P}_1 = 0 \tag{4.4}$$

and choose "starting points" as nonzero solutions of $e_1(P_1,P_2)$ and $e_2(P_1,P_2)$

$$e_1(P_1,P_2) = P_1(A + \alpha I) + (A + \alpha I)^TP_1$$
$$- P_1B_1R_{11}^{-1}B_1^TP_1 - P_1B_2R_{22}^{-1}B_2^TP_1 = 0, \tag{4.5}$$

$$e_2(P_1,P_2) = P_2(A + \alpha I) + (A + \alpha I)^TP_2$$
$$- P_2B_1R_{11}^{-1}B_1^TP_2 - P_2B_2R_{22}^{-1}B_2^TP_2 = 0 \tag{4.6}$$

where α is either zero or a positive random number plus the negation of smallest non-positive eigenvalues of the matrix A. The reason to choose such an α is the same as described in Section 3.3.1. The coupled homotopy equations become

$$h_1(P_1,P_2) = e_1(P_1,P_2) + t(f_1(P_1,P_2) - e_1(P_1,P_2))$$
$$= P_1(A + \alpha I) + (A + \alpha I)^TP_1$$

$$- P_1B_1R_{11}^{-1}B_1^TP_1 - P_1B_2R_{22}^{-1}B_2^TP_1$$

$$+ t\ [Q_1 - 2aP_1 - P_1B_2R_{22}^{-1}B_2^TP_2 - P_2B_2R_{22}^{-1}B_2^TP_1$$

$$+ P_1B_2R_{22}^{-1}B_2^TP_1 + P_2B_2R_{22}^{-1}R_{12}R_{22}^{-1}B_2^TP_2] = 0 \qquad (4.7)$$

$$h_2(P_1,P_2) = e_2(P_1,P_2) + t(f_2(P_1,P_2) - e_2(P_1,P_2))$$

$$= P_2(A + aI) + (A + aI)^TP_2$$

$$- P_2B_1R_{11}^{-1}B_1^TP_2 - P_2B_2R_{22}^{-1}B_2^TP_2$$

$$+ t\ [Q_2 - 2aP_2 - P_2B_1R_{11}^{-1}B_1^TP_1 - P_1B_1R_{11}^{-1}B_1^TP_2$$

$$+ P_2B_1R_{11}^{-1}B_1^TP_2 + P_1B_1R_{22}^{-1}R_{21}R_{11}^{-1}B_1^TP_1] = 0 \qquad (4.8)$$

Differentiating both Eqs. (4.7) and (4.8) yield the following homotopy differential equations (or Davidenko differential equation)

$$\dot{P}_1h_{1_{P_1}} + h_{1_{P_1}}^T\dot{P}_1 + \dot{P}_2h_{1_{P_2}} + h_{1_{P_2}}^T\dot{P}_2 = h_{1_t} \qquad (4.9)$$

$$\dot{P}_1h_{2_{P_1}} + h_{2_{P_1}}^T\dot{P}_1 + \dot{P}_2h_{2_{P_2}} + h_{2_{P_2}}^T\dot{P}_2 = h_{2_t} \qquad (4.10)$$

where $h_{i_{P_j}} \equiv \dfrac{\partial h_i}{\partial P_j}(i,j = 1,2)$ and $h_{i_t} \equiv \dfrac{\partial h_i}{\partial t}(i = 1,2)$. These quantities are:

$$h_{1_{P_1}} = [A - B_1R_{11}^{-1}B_1^TP_1 + (1 - t)(aI - B_2R_{22}^{-1}B_2^TP_1) - t\ B_2R_{22}^{-1}B_2^TP_2]^T$$

$$h_{1_{P_2}} = t[B_2R_{22}^{-1}R_{12}R_{22}^{-1}B_2^TP_2 - B_2R_{22}^{-1}B_2^TP_1]^T$$

$$h_{2_{P_1}} = t[B_1R_{11}^{-1}R_{21}R_{11}^{-1}B_1^TP_1 - B_1R_{11}^{-1}B_1^TP_2]^T$$

$$h_{2_{P_2}} = [A - B_2R_{22}^{-1}B_2^TP_2 + (1 - t)(aI - B_1R_{11}^{-1}B_1^TP_2) - t\ B_1R_{11}^{-1}B_1^TP_1]^T$$

$$h_{1_t} = 2aP_1 + P_2B_2R_{22}^{-1}B_2^TP_1 + P_1B_2R_{22}^{-1}B_2^TP_2$$

$$- P_2B_2R_{22}^{-1}R_{12}R_{22}^{-1}B_2^TP_2 - P_1B_2R_{22}^{-1}B_2^TP_1 - Q_1$$

$$h_{2_t} = 2aP_2 + P_1B_1R_{11}^{-1}B_1^TP_2 + P_2B_1R_{11}^{-1}B_1^TP_1$$

$$- P_1B_1R_{11}^{-1}R_{21}R_{11}^{-1}B_1^TP_1 - P_2B_1R_{11}^{-1}B_1^TP_2 - Q_2$$

Now, define H and $Vec(P)$ as

$$H = \begin{Bmatrix} Vec(h_1) \\ Vec(h_2) \end{Bmatrix} \tag{4.11}$$

$$Vec(P) = \begin{Bmatrix} Vec(P_1) \\ Vec(P_2) \end{Bmatrix} \tag{4.12}$$

where $Vec(P_1)$ and $Vec(P_2)$ are

$$Vec(P_1) = [p_{1_{11}}\ p_{1_{21}}\ \cdots\ p_{1_{m1}} | p_{1_{12}}\ p_{1_{22}}\ \cdots\ p_{1_{m2}} | \ldots\ \ldots\ \ldots\ \ldots | p_{1_{1n}}\ p_{1_{2n}}\ \cdots\ p_{1_{mn}}]^T$$

$$Vec(P_2) = [p_{2_{11}}\ p_{2_{21}}\ \cdots\ p_{2_{m1}} | p_{2_{12}}\ p_{2_{22}}\ \cdots\ p_{2_{m2}} | \ldots\ \ldots\ \ldots\ \ldots | p_{2_{1n}}\ p_{2_{2n}}\ \cdots\ p_{2_{mn}}]^T$$

Notice that Eqs. (4.9) and (4.10) are in the form of coupled Lyapunov equations, which cannot be solved with standard Lyapunov solvers. In order to solve these coupled Lyapunov equations, a vectorization procedure similar to that described in Chapter 3 has to be performed [6,49]. Equations (4.9) and (4.10) must be vectorized and are rewritten in the matrix equation form:

$$\begin{bmatrix} h_{1_{P_1}} \oplus h_{1_{P_1}} & h_{1_{P_2}} \oplus h_{1_{P_2}} \\ h_{2_{P_1}} \oplus h_{2_{P_1}} & h_{2_{P_2}} \oplus h_{2_{P_2}} \end{bmatrix} \frac{d\ Vec(P)}{d\ t} = \begin{Bmatrix} Vec(h_{1_t}) \\ Vec(h_{2_t}) \end{Bmatrix} \tag{4.13}$$

Hence, solving the coupled algebraic Riccati equations (Eqs. (4.1) and (4.2)) is equivalent to finding the solution of Eq. (4.13) with two sets of starting points from non-zero solutions of the following uncoupled Lyapunov equations

$$(A + aI)P_1^{-1} + P_1^{-1}(A + aI)^T - B_1R_{11}^{-1}B_1^T - B_2R_{22}^{-1}B_2 = 0 \tag{4.14}$$

$$(A + aI)P_2^{-1} + P_2^{-1}(A + aI)^T - B_1R_{11}^{-1}B_1^T - B_2R_{22}^{-1}B_2 = 0 \tag{4.15}$$

This derivation and procedure of the proposed homotopy algorithm for coupled algebraic Riccati equations is similar to the one for the algebraic Riccati equation that was presented in Chapter 3. Both algorithms starts with the solution of a Lyapunov-type equation, and then integrates the Lyapunov-type differential equation over the homotopy variable *t*. However, the homotopy algorithm for the solution of coupled algebraic Riccati equations involves the integration of coupled differential equations. This coupling effect

complicates the analysis on the positive definiteness of the solution. Thus, unlike the algorithm presented in Chapter 3, this proposed homotopy algorithm was not analyzed for the positive definiteness, hence there is no guarantee that the solution to the coupled algebraic Riccati equations will be positive definite.

The proposed homotopy algorithm is more efficient than the generic homotopy codes. This improvement in efficiency results from using the symmetric property of the coupled algebraic Riccati equations in conjunction with the Kronecker sum. Also, as was shown in Chapter 3, the "integration path" will "never" bifurcate (i.e., bifurcation points can be bypassed along the homotopy path), according to the Sard's theorem (Theorem 3.2.3 and Corollary 3.2.3).

4.3 Numerical Examples

Two examples are presented to demonstrate the computational efforts vs. convergence guarantee for solving the two-person non-cooperative non-zero sum differential game problems.

Model:

Given a dynamical system as follow

$$\dot{x} = Ax + B_1 u_1 + B_2 u_2$$

Find a pair of feedback controllers, u_1 and u_2, to minimize the following performance indices simultaneously

$$\min_{u_j} \quad J_j = \frac{1}{2}\int_0^{\infty} (x_1^T Q_j x + u_1^T R_{j1} u_1 + u_2^T R_{j2} u_2)\, dt, \quad j = 1,2$$

where

$$u_j = -R_{jj}^{-1} B_j^T P_j x, \; j = 1,2$$

Example 4.1. The example is selected from an isolation problem [6].

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -10.0040 & 0.0040 & -0.0316 & 0.0000 \\ 0.4000 & -0.4000 & 0.0013 & -0.0013 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 0 \\ 0 \\ 0.0100 \\ 0 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 \\ 0 \\ -0.0100 \\ 1.0000 \end{bmatrix}$$

$$Q_1 = Diag([100 \quad 0 \quad 1 \quad 0]), \; Q_2 = Diag([0 \quad 100 \quad 0 \quad 1])$$

$$R_{11} = 1, \; R_{12} = 0, \; R_{21} = 0, \; R_{22} = 1$$

Both the Newton-iterative scheme [1] and the proposed homotopy scheme are used to solve this example. Both schemes lead to the same positive definite solution pair

$$P_1 = \begin{bmatrix} 1.4200 & 0.0001 & 0.0050 & -0.0031 \\ 0.0001 & 0.0001 & 0.0031 & 0.0000 \\ 0.0050 & 0.0031 & 0.1420 & 0.0014 \\ -0.0031 & 0.0000 & 0.0014 & 0.0000 \end{bmatrix} \times 10^3$$

$$P_2 = \begin{bmatrix} 0.9654 & 0.8461 & 0.0080 & 0.3952 \\ 0.8461 & 44.9974 & -0.0111 & 9.6079 \\ 0.0080 & -0.0111 & 0.0966 & 0.0868 \\ 0.3952 & 9.6079 & 0.0868 & 4.4958 \end{bmatrix}$$

The Newton-iterative scheme has a stopping criterion that 2-norm of both $f_1(P_1, P_2)$ and $f_2(P_1, P_2)$ are less than 10^{-7}, where $f_1(P_1, P_2)$ and $f_2(P_1, P_2)$ are defined in Eqs. (4.3) and (4.4), respectively. By using the proposed homotopy algorithm, the 2-norm of $f_1(P_1, P_2)$ is 1.6307×10^{-7} and the 2-norm of $f_2(P_1, P_2)$ is 1.5231×10^{-6}. The scheme based on the Newton-iterative method takes 227,838 flops (using "flops" command in MATLAB) to reach the solution, and the "efficient" homotopy scheme takes 46.8228 mega flops. (The other homotopy scheme, which does not use the symmetric property of the Riccati solutions, takes 234.6111 mega flops.) Notice that mega flops is the unit used when evaluating the floating points operation for a homotopy scheme (e.g., see Reference 50). This example shows that if the iterative scheme converges, the iterative scheme is typical more efficient

than the homotopy scheme. To demonstrate a convergence failure of the Newton-iterative scheme, we examine the next example.

Example 4.2. This example shows that only the homotopy scheme yields solutions.

$$A = \begin{bmatrix} 2 & 1 & 0 & 1 \\ 1 & 2 & 1 & 0 \\ 0 & 1 & 2 & 1 \\ 1 & 0 & 1 & 2 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 1 & 1 \\ 1 & 1 \end{bmatrix},$$

$$Q_1 = Diag([1 \quad 0 \quad 1 \quad 0]), \; Q_2 = Diag([0 \quad 1 \quad 0 \quad 1]),$$

$$R_{11} = Diag([1 \;\; 1]), \; R_{12} = Diag([0 \;\; 0]), \; R_{21} = Diag([0 \;\; 0]), \; R_{22} = Diag([1 \;\; 1]).$$

Both Newton-iterative scheme [1] and the proposed homotopy scheme are used to solve this example. However, only the homotopy scheme yields the solution pair,

$$P_1 = \begin{bmatrix} 4.5497 & 0.9949 & -0.7060 & 1.2931 \\ 0.9949 & 0.8287 & 0.7205 & -0.7285 \\ -0.7060 & 0.7205 & 1.8938 & -1.9439 \\ 1.2931 & -0.7285 & -1.9439 & 2.2724 \end{bmatrix}$$

$$P_2 = \begin{bmatrix} 0.4732 & 1.2043 & -0.8321 & 0.6433 \\ 1.2043 & 3.8848 & -2.7721 & 0.9137 \\ -0.8321 & -2.7721 & 2.2196 & -0.1944 \\ 0.6433 & 0.9137 & -0.1944 & 2.2560 \end{bmatrix}$$

the Newton-iterative scheme never converges to the solution. The homotopy scheme takes 36.1058 mega flops to get the solution. (The other homotopy scheme, which does not use the symmetric property of the Riccati solutions, takes 195.2124 mega flops.) The 2-norm of f_1 is 1.4817×10^{-9} and the 2-norm of f_2 is 2.3112×10^{-9}.

4.4 Summary

This chapter introduces a new homotopy scheme to solve strongly coupled algebraic Riccati equations. The proposed algorithm is more efficient than a generic homotopy scheme, since it utilize the symmetric property of solutions of Riccati matrix in conjunction with the Kronecker sum. The convergence of the homotopy algorithm to a solution of the

coupled algebraic Riccati equations is guaranteed by the Sard's Theorem. Two numerical examples are given and solutions attempted by both iterative (Newton) and integrative (homotopy) schemes. Only the homotopy scheme was successfully in both examples. It should be noted that in general integrative schemes require more floating point operation than iterative schemes. However, unlike the iterative scheme which may fail to converge to a solution, the integrative homotopy scheme is guaranteed to yield a solution.

Although the developed algorithm is capable of solving strongly coupled algebraic Riccati equations, its utilization in solving real control problems may be limited. These limitations in application results from (1) too much computational effort is required to design state feedback controllers, even for the 2-player case, and (2) it is very difficult to choose input and state weighting matrices in the performance indices to satisfy system design criteria (e.g., settling time, overshoot of the state time response, etc.). In Chapters 5 and 6, an inverse procedure (i.e., given desired closed-loop poles and input weighting matrices, determine the state weighting matrices and the feedback gain matrices) is proposed to overcome the above limitations.

CHAPTER 5
DESIGN OF UNCONSTRAINED AND CONSTRAINED GAME THEORETIC CONTROLLERS WITH PRESCRIBED EIGENVALUES

5.1 Introduction

Current researches on solving the differential game control have all concentrated on determining feedback gain matrices for the 2-player differential game with given state and control weighting matrices (i.e., solving a system of 2-coupled algebraic Riccati equation). For a general n-player differential game, a solution scheme is still not available due to the complexity and difficulty in solving n-coupled algebraic Riccati equations numerically. In addition, based on these approach, there is no systematic method for obtaining desired pole locations for the closed-loop system.

Since the settling time of a stable linear system is strongly dependant on the location of the poles of the system, pole placement via state feedback controllers has been an active area of research. For example, Bass and Gura [51] used a determinant identity technique to obtain feedback gains such that a single input linear system has desired eigenvalues. Ackermann [52] proposed a pole placement technique for a multiple input linear system by utilizing the "Ackermann's formula" (i.e., $k = q_n^T\alpha(A)$, where k is the state feedback gain, q_n^T is the last row of the controllability matrix, α_i is the coefficients of closed-loop characteristic polynomial equation, and $\alpha(A) \equiv A^n + \alpha_1 A^{n-1} + \ldots + \alpha_n I$). By utilizing a pseudo inversion technique, Andry, Shapiro, and Chung [53] were able to assign the eigenstructure of a multiple input linear system .

By selection of various state feedback controllers, it is possible that the resulting closed-loop systems will have identical sets of the closed-loop poles; however, each

controller may result in an entirely different transient response (e.g., see numerical examples in Chapter 6). The methodology of "optimal pole placement" which ties pole placement with optimal control was developed to choose the "optimal controller" among those designations for the linear regulator problem. Optimal pole placement was first modeled by Kalman [54] as follow "given a control law, find all the performance indices for which this control law is optimal." Numerous researches, based on Kalman's original definition, have been developed to design linear quadratic regulators with a prescribed set of eigenvalues. For example, given the control weighting matrix and the desired real part of the closed-loop pole, Solheim [55], Graupe [56], and Amin [57] presented different methods to find state weighting and feedback matrices to form a closed-loop system. Later, Saif [58], Arar and Sawan [59] proposed schemes which move a pair of complex open-loop poles to another pair of pole location (a pair of complex or two real poles) for a closed-loop system.

Current optimal pole placement approaches are based on the assumption that the state feedback controllers optimize a linear system with a single objective. In this chapter, we propose a new optimal pole placement methodology based on differential game theory for the general multiple objective control problems. By using multiple objective functions, it can expand the control domain and thus allow the control system designer the freedom of implementing additional performance criteria. Furthermore, to the best of this author's knowledge, the issue of weighting matrices selection for a given set of closed-loop poles in linear quadratic differential games has never been addressed in the literature. The method proposed in this chapter is capable of shifting either a single pole or a pair of poles (two real poles or a pair of complex conjugate poles) to the desired location(s) while satisfying additional design specifications for the linear system. Multiple poles shifting tasks can be implemented by utilizing the shifting algorithm for a single pole or a pair of poles recursively.

There are several advantages (over the conventional approach to the classical optimal control problems and the differential games) of the proposed methodology [60]:

(1). The proposed scheme can determine state feedback controllers in a 2-player differential game without actually solving 2-coupled algebraic Riccati equations numerically.

(2). The proposed scheme can easily be extended to solve the general n-player differential game.

(3). The proposed scheme can incorporate an additional design criterion superimposed as an optimization criterion, e.g., the minimum of the square of the Frobenius norm of the state feedback gain matrix can be implemented by solving a nonlinear programing problem (for detail see Chapter 6).

(4). The solution of the game theoretic controller is proven to be at least equal to or better than the solution of the LQR controller, since the latter one belongs to the solution set of the former one (detail for a single pole shifting case see Theorem 5.1 in Section 5.3.2).

(5). The proposed scheme can also incorporate other physical design constraints as additional inequality constraints for this optimization problem, e.g., the limitations which may arise from vibration suppression, based on the consideration of both comfortability and performance, on a vehicle design (for detail see Chapter 6).

This chapter is organized in the following manner. The principles of optimum pole placement for single objective control problems is presented in section 2. Section 3 extends the principles to multiobjective control problems. Section 4 summarizes the proposed pole shifting scheme. The issue of additional criteria that must be superimposed in the case of pole placement for multiobjective control problems will be discussed in Chapter 6.

5.2 Review of an Optimal Pole Placement for Single Objective Control Problem

This short review is based on the recent research by Arar [59]. A detail presentation of this work along with a correction of the optimization criterion for the case of shifting a pair of poles are also presented in the Appendix.

Given a linear controllable and observable dynamical system

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{5.1}$$

where $u(t) \in \Re^m$ represents control input to the plant, and $x(t) \in \Re^n$ represents the perfectly measured plant state. Based on the performance index

$$J = \int_0^\infty (x^T Q x + u^T R u)\, dt \tag{5.2}$$

the state feedback controller is

$$u(t) = -R^{-1}B^T P x(t) \tag{5.3}$$

where P is the positive definite solution of the following algebraic Riccati equation

$$Q + PA + A^T P - PBR^{-1}B^T P = 0 \tag{5.4}$$

Consider the following transformation

$$x_r = T^T x \tag{5.5}$$

where $x \in \Re^n$ represents the plant states of the "true" model (i.e., the full plant states), $x_r \in \Re^r$ represents the "reduced" plant states, and $T \in \Re^{n \times r}$ is the set of left eigenvectors of the open-loop system matrix A associated with the open-loop poles which are to be shifted. Applying this transformation to Eq. (5.1) yields a system of equations which is referred as the "reduced" model [61]

$$\dot{x}_r(t) = A_r x_r(t) + B_r u_r(t) \tag{5.6}$$

where $A_r \in \Re^{r \times n}$ and $B_r \in \Re^{r \times m}$ are

$$T^T A = A_r T^T \tag{5.7}$$

$$B_r = T^T B \tag{5.8}$$

The reduced-order state feedback control law, u_r, of the reduced system is

$$u_r(t) = -R^{-1}B_r^T P_r x_r(t) \tag{5.9}$$

P_r is the positive definite solution of the reduced algebraic Riccati equation

$$Q_r + P_r A_r + A_r^T P_r - P_r B_r R^{-1} B_r^T P_r = 0 \tag{5.10}$$

where

$$P = TP_rT^T \tag{5.11}$$

The reduced-order state feedback gain matrix is

$$K_r = R^{-1}B_rP_r \tag{5.12}$$

which results in the reduced-order closed-loop system

$$\dot{x}_r(t) = (A_r - B_rK_r)x_r(t) \tag{5.13}$$

Observe that the characteristic polynomial equation for the reduced-order closed-loop system $A_r - B_rK_r$ has the poles at their desired closed-loop locations. For a single pole shifting case, let the characteristic polynomial equation for the reduced-order closed-loop system be

$$s - a_0(\alpha, B_r, R, P_r) = 0 \tag{5.14}$$

where the quantity α is the open-loop pole to be shifted. Assume that this open-loop pole moves from its current location to a new location, say μ, then the characteristic polynomial equation for the reduced-order closed-loop system can also be expressed as

$$s - \mu = 0 \tag{5.15}$$

By comparing coefficients between Eqs. (5.14) and (5.15), P_r is obtained by solving

$$\mu = a_0(\alpha, B_r, R, P_r) \tag{5.16}$$

Similarly for a pair of poles shifting case, let the characteristic polynomial equation for the reduced-order closed-loop system to be

$$s^2 - a_1(\alpha_1, \alpha_2, B_r, R, P_r)s + a_0(\alpha_1, \alpha_2, B_r, R, P_r) = 0 \tag{5.17}$$

where the quantities α_1 and α_2 are the open-loop poles to be shifted. Suppose that these open-loop poles move from their current location to new locations, say μ_1 and μ_2, then the characteristic polynomial equation for the reduced-order closed-loop system can also be written as

$$s^2 - (\mu_1 + \mu_2)s + \mu_1\mu_2 = 0 \tag{5.18}$$

By comparing coefficients between Eqs. (5.17) and (5.18), P_r is obtained by solving

$$\mu_1 + \mu_2 = a_1(\alpha_1, \alpha_2, B_r, R, P_r)$$

$$\mu_1\mu_2 = -a_0(\alpha_1, \alpha_2, B_r, R, P_r)$$

5.3 Pole Placement for Multiobjective Problem

5.3.1 Preliminary Transformation

For conciseness, the following derivation is based on the 2-player differential game. The n-player differential game problem is easily derived in a similar manner. (An example is given in Chapter 6.) Consider the following transformation

$$x_r = T^T x \tag{5.19}$$

where x, x_r and T are defined in Section 5.2. Applying this transformation to Eq. (2.1) yields a system of equations which is also referred as the "reduced" model

$$\dot{x}_r(t) = A_r x_r(t) + B_{1_r} u_{1_r}(t) + B_{2_r} u_{2_r}(t) \tag{5.20}$$

where $A_r \in \Re^{r \times r}$ satisfies $T^T A = A_r T^T$, $B_{1_r} = T^T B_1 \in \Re^{r \times m_1}$, $B_{2_r} = T^T B_2 \in \Re^{r \times m_2}$.

The performance indices associated with the reduced system are

$$J_{i_r}(u_{1_r}, u_{2_r}) = \int_0^{\infty} (x_r^T Q_{i_r} x_r + u_{1_r}^T R_{i1} u_{1_r} + u_{2_r}^T R_{i2} u_{2_r})\ dt, \quad i = 1,2 \tag{5.21}$$

where $Q_{i_r} \in \Re^{r \times r}\ (i = 1,2)$ is related to the $Q_i\ (i = 1,2)$ in Eq. (2.2) by

$$Q_i = T Q_{i_r} T^T, \ i = 1,2 \tag{5.22}$$

and the control weighting matrices R_{i1} and $R_{i2}\ (i = 1,2)$ remains the same as in Eq. (2.2). Assuming the state feedback for the reduced system results in the following control actions

$$u_{i_r}(t) = -R_{ii}^{-1} B_{i_r}^T P_{i_r} x_r(t), \ i = 1,2 \tag{5.23}$$

where P_{i_r}, $i = 1,2$, are the positive definite solutions of the coupled algebraic Riccati equations associated with the reduced model

$$\begin{aligned} Q_{1_r} + P_{1_r} A_r + A_r^T P_{1_r} - P_{1_r} B_{1_r} R_{11}^{-1} B_{1_r}^T P_{1_r} - P_{1_r} B_{2_r} R_{22}^{-1} B_{2_r}^T P_{2_r} & \\ - P_{2_r} B_{2_r} R_{22}^{-1} B_{2_r}^T P_{1_r} + P_{2_r} B_{2_r} R_{22}^{-1} R_{12} R_{22}^{-1} B_{2_r}^T P_{2_r} &= 0 \end{aligned} \tag{5.24}$$

$$Q_{2_r} + P_{2_r} A_r + A_r^T P_{2_r} - P_{2_r} B_{2_r} R_{22}^{-1} B_{2_r}^T P_{2_r} - P_{2_r} B_{1_r} R_{11}^{-1} B_{1_r}^T P_{1_r}$$

$$- P_{1_r}B_{1_r}R_{11}^{-1}B_{1_r}^TP_{2_r} + P_{1_r}B_{1_r}R_{11}^{-1}R_{21}R_{11}^{-1}B_{1_r}^TP_{1_r} = 0 \quad (5.25)$$

where

$$P_i = TP_{i_r}T^T, \; i = 1,2 \quad (5.26)$$

The closed-loop system for the reduced model is

$$\dot{x}_r(t) = (A_r - \sum_{i=1}^{2} B_{i_r}R_{ii}^{-1}B_{i_r}^TP_{i_r})x_r(t) \quad (5.27)$$

For future discussion, we define the feedback gain matrix for the reduced system as

$$K_r = \begin{bmatrix} R_{11}^{-1}B_{1_r}^TP_{1_r} \\ R_{22}^{-1}B_{2_r}^TP_{2_r} \end{bmatrix} \quad (5.28)$$

The resulting closed-loop system equation for the true states can be written as

$$\dot{x}(t) = (A - \sum_{i=1}^{2} B_iR_{ii}^{-1}B_{i_r}^TP_{i_r}T^T)x(t) \quad (5.29)$$

Similarly, we define the feedback gain matrix for the "true" system as

$$K = K_rT^T = \begin{bmatrix} R_{11}^{-1}B_{1_r}^TP_{1_r} \\ R_{22}^{-1}B_{2_r}^TP_{2_r} \end{bmatrix} T^T \quad (5.30)$$

It has been shown by Rao and Lamba [61] that the n closed-loop poles of Eq. (5.29) consists of the r closed-loop poles of Eq. (5.27) and the $n - r$ undisturbed open-loop poles of Eq. (2.1). We now present the procedure for shifting a single pole and a pair of poles.

5.3.2 Shifting a Single Pole

Let α be a single pole of the linear system which is to be moved from its current location to a new location, say μ. From Eqs. (5.19), (5.20) and (5.23), and choosing T to be the left eigenvector associated with the left eigenvalue, we get

$$\mu = \alpha - B_{1_r}R_{11}^{-1}B_{1_r}^TP_{1_r} - B_{2_r}R_{22}^{-1}B_{2_r}^TP_{2_r} \quad (5.31)$$

Notice that Eq. (5.31) is an underdetermined system (i.e., there are fewer equations than unknowns), thus the solution of the Eq. (5.31) is not unique. The non-uniqueness of choice

for P_{1_r} and P_{2_r} gives the control system engineer the ability to implement design specifications, such as minimum envelop bound of the system transient response, as addition optimality criteria. On the contrary, the classical "inverse optimal" schemes, (e.g., see References 57 and 59), do not provide such a design space. Several valid optimization criteria are presented in the section 4. Notice that the inequality constraints for the optimization problem are $P_{i_r} > 0,\ (i = 1, 2)$.

Comparison with a LQR controller

For a LQR controller, a and μ are the associated open-loop and closed-loop poles, respectively, for a game theoretic controller. Using the results from Arar and Sawan [59] (modified version with correction provided in the Appendix), the reduced-order closed-loop system matrix associated with the LQR controller satisfies the equality constraint equation

$$\mu = a - \left\{ B_{1_r}\ B_{2_r} \right\} \begin{Bmatrix} R_{11}^{-1} & 0 \\ 0 & R_{22}^{-1} \end{Bmatrix} \begin{Bmatrix} B_{1_r}^T \\ B_{2_r}^T \end{Bmatrix} P_r \tag{5.32}$$

Expanding Eq. (5.32) yields

$$\mu = a - B_{1_r} R_{11}^{-1} B_{1_r}^T P_r - B_{2_r} R_{22}^{-1} B_{2_r}^T P_r \tag{5.33}$$

Therefore, the solution to the reduced algebraic Riccati equation is

$$P_r = \frac{a - \mu}{B_{1_r} R_{11}^{-1} B_{1_r}^T + B_{2_r} R_{22}^{-1} B_{2_r}^T} \tag{5.34}$$

For a process which is stabilizing, (i.e., moving a system pole to its left in the s-plane), it can be shown that the solution for the LQR controller is a subset of the solution of the game theoretic controller in a single pole assignment problem. The following theorem and a numerical example are presented to reinforce this premise.

Theorem 5.1

For the case that a single pole is shifted to its left from its current location in the s-plane (i.e., the closed-loop pole is more negative than the open-loop pole), the solution of the LQR

controller (Eq. (5.34)) belongs to the solution set of the game theoretic controller (Eq. (5.31)).

Proof:

For the case of interest, $\alpha > \mu$. Thus P_r is positive, since $\alpha - \mu$ is positive and $B_{1_r}R_{11}^{-1}B_{1_r}^T + B_{2_r}R_{22}^{-1}B_{2_r}^T$ are positive for any non-zero B $(= [\ B_1\ B_2\ ])$. By substituting P_r for P_{1_r} and P_r for P_{2_r} into Eq. (5.31) yields

$$\mu = \alpha - B_{1_r}R_{11}^{-1}B_{1_r}^TP_r - B_{2_r}R_{22}^{-1}B_{2_r}^TP_r$$

which is identical to Eq. (5.33). Hence, the solution of the LQR controller (Eq. (5.34)) belongs to the solution set of the game theoretic controller (Eq. (5.31)).

❑

In a 2-player game, the solution set of the game theoretic controller is a straight line in the first quadrant of $P_{1_r}P_{2_r}$ plane, and the solution of the LQR controller is a point on that straight line (see Example 5.1 below).

Example 5.1:

Consider the following lateral model of the F-4 aircraft [62]. The controllable system is given as

$$\dot{x} = \begin{bmatrix} -1.7680 & 0.4125 & -14.2500 & 0.0000 \\ -0.0070 & -0.3831 & 6.0380 & 0.0000 \\ 0.0016 & -0.9975 & -0.1551 & 0.0586 \\ 1.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix} x + \begin{bmatrix} 1.7440 & 8.9520 \\ -2.9200 & -0.3075 \\ 0.0243 & -0.0036 \\ 0.0000 & 0.0000 \end{bmatrix} u$$

where

$$x = \begin{Bmatrix} p \\ r \\ \beta \\ \phi \end{Bmatrix} = \begin{Bmatrix} \text{roll rate} \\ \text{yaw rate} \\ \text{sideslip} \\ \text{roll angle} \end{Bmatrix} \quad \text{and} \quad u = \begin{Bmatrix} \delta_r \\ \delta_a \end{Bmatrix} = \begin{Bmatrix} \text{rudder} \\ \text{aileron} \end{Bmatrix}$$

The quadratic performance indices are

$$J_i = \int_0^{\infty} (x^T Q_i x + u_1^T R_{i1} u_1 + u_2^T R_{i2} u_2)\ dt \quad (i = 1, 2)$$

where $R_{11} = 1$, $R_{12} = 0$, $R_{21} = 0$, and $R_{22} = 1$. Partition the control task results in

$$B_1 = [1.7440 \ -2.9200 \quad 0.0243 \quad 0.0000]^T$$

$$B_2 = [8.9520 \ -0.3075 \ -0.0036 \quad 0.0000]^T$$

In order to compare the game theoretic state feedback controller with a LQR state feedback controller, the following performance index is chosen for the LQR controller

$$J = J_1 + J_2 = \int_0^{\infty} (x^T Q x + u^T R u)\ dt$$

where the state weighting matrix Q and the control weighting matrix R are

$$Q = Q_1 + Q_2, \quad R = diag([R_{11} \ R_{22}])$$

The open-loop poles for the above system matrix of the F-4 aircraft lateral model are located at $[-0.0150, \ -0.2149 \pm 2.4858j, \ -1.8614]$. These poles correspond to the spiral mode, the Dutch roll mode, and the roll mode, respectively. Assume that the control task is to move the first open-loop pole of the system from its current location to a new location at -0.5. The solution set to the game theoretic controller and the solution to the LQR controller are shown in Figure 5.1. In Figure 5.1, it is clear that the solution to the LQR controller belongs to the solution set to game theoretic controller for single pole shifting case. Notice that if the compatible performance index is chosen differently, i.e.,

$$J = a_1 J_1 + a_2 J_2 = \int_0^{\infty} (x^T Q x + u^T R u)\ dt, \qquad a_i \neq 1 \ (i = 1, 2),$$

the solution to the LQR controller is not the same point shown in Figure 5.1. However, the solution to the LQR controller still lies on the line segment, which is the solution set to the game theoretic controller.

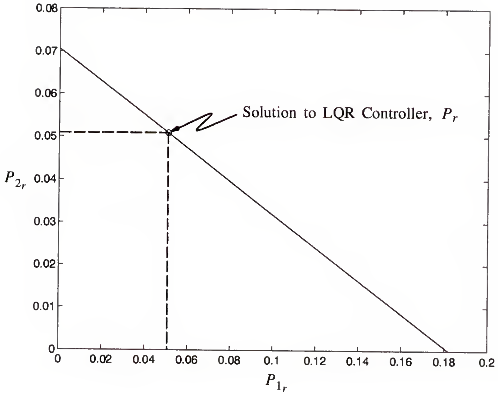

Fig. 5.1 The solution to LQR controller belongs to the solution set to game theoretic controller

5.3.3 Shifting a Pair of Poles

Let $\alpha + \beta j,\ \xi - \beta j$ be a pair of distinct left eigenvalues of an $n \times n$ system matrix A which are to be shifted from their current location to a new location at $\gamma + \delta j, \phi - \delta j$. Also, let the associated nonzero left eigenvectors be $u \pm jv$ (or u and v for two real poles). For the open-loop poles,

$$\xi = \alpha, \ \text{if}\ \beta \neq 0 \ \text{(a pair of complex conjugate poles)}$$

$$\xi \neq \alpha, \ \text{if}\ \beta = 0 \ \text{(two real, distinct poles)}$$

and for the closed-loop poles,

$$\phi = \gamma, \ \text{if}\ \delta \neq 0 \ \text{(a pair of complex conjugate poles)}$$

$$\phi \neq \gamma, \ \text{if}\ \delta = 0 \ \text{(two real, distinct poles)}$$

The following definition is helpful for the case of shifting complex eigenvalues.

Definition 5.1 Van Loan [63]

Let A be a $n \times n$ matrix. A scalar, λ, is defined as a left eigenvalue of the matrix A with an associated nonzero left eigenvector y if the following condition holds

$$y^H A = \lambda y^H$$

❑

Now, let $\alpha \pm \beta j$ be the complex conjugate pair of left eigenvalues of the system matrix A, with associated nonzero left eigenvectors, $u \pm jv$ $(u, v \in \Re^n)$. Based on the Definition 5.1, then

$$\begin{bmatrix} u^T - jv^T \\ u^T + jv^T \end{bmatrix} A = \begin{bmatrix} \alpha + \beta j & 0 \\ 0 & \alpha - \beta j \end{bmatrix} \begin{bmatrix} u^T - jv^T \\ u^T + jv^T \end{bmatrix} \tag{5.35}$$

To obtain the "real form" of the left eigensolution equation [64], Eq. (5.35) is pre-multiplied by a transformation matrix, $\mathcal{T}$

$$\mathcal{T} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2}j & -\frac{1}{2}j \end{bmatrix} \tag{5.36}$$

which yields

$$\begin{Bmatrix} u^T \\ v^T \end{Bmatrix} A = \Lambda^* \begin{Bmatrix} u^T \\ v^T \end{Bmatrix} \tag{5.37}$$

where

$$\Lambda^* = \mathcal{T} \begin{bmatrix} \alpha + \beta j & 0 \\ 0 & \alpha - \beta j \end{bmatrix} \mathcal{T}^{-1} = \begin{bmatrix} \alpha & \beta \\ -\beta & \alpha \end{bmatrix} \tag{5.38}$$

Eq. (5.37) is called the "real form" of left eigensolution equation. Notice that Eq. (5.37) was used to define the state matrix for the reduced model. Thus, the required transformation matrix is

$$T = \begin{Bmatrix} u^T \\ v^T \end{Bmatrix}$$

In the case of shifting two distinct real poles, the transformation matrix, T, has the same form as shown above. However, u and v are the left eigenvectors of the system matrix corresponding to the two distinct real eigenvalues.

Returning to the case of shifting a pair of poles, application of the above transformation matrix to Eq. (2.1) yields

$$\dot{x}_r(t) = A_{r_c}\, x_r(t)$$

where the definitions in Eqs. (5.19), (5.20) and (5.23) were used to define the closed-loop system matrix A_{rc}, as

$$A_{r_c} = A_r - \sum_{i=1}^{2} B_{i_r} R_{ii}^{-1} B_{i_r}^T P_{i_r} \tag{5.39}$$

The matrix definitions in Eq. (5.39) are

$$A_r = \begin{bmatrix} \alpha & \beta \\ -\beta & \xi \end{bmatrix} \tag{5.40}$$

$$P_{1_r} = \begin{bmatrix} x_1 & z_1 \\ z_1 & y_1 \end{bmatrix} \tag{5.41}$$

$$P_{2_r} = \begin{bmatrix} x_2 & z_2 \\ z_2 & y_2 \end{bmatrix} \tag{5.42}$$

In the system matrix A_r of the reduced model, $\xi = \alpha$ and $\beta \neq 0$ for the case of the complex conjugate pair of poles, and $\xi \neq \alpha$ and $\beta = 0$ for the case of the real distinct poles. Now, let

$$B_{1_r} R_{11}^{-1} B_{1_r}^T = \begin{bmatrix} a_1 & b_1 \\ b_1 & c_1 \end{bmatrix} \tag{5.43}$$

$$B_{2_r} R_{22}^{-1} B_{2_r}^T = \begin{bmatrix} a_2 & b_2 \\ b_2 & c_2 \end{bmatrix} \tag{5.44}$$

Substitution of A_r, P_{1_r}, P_{2_r}, $B_{1_r} R_{11}^{-1} B_{1_r}^T$, and $B_{2_r} R_{22}^{-1} B_{2_r}^T$ into Eq. (5.39) yields

$$A_{r_c} = \begin{bmatrix} \alpha - a_1x_1 - b_1z_1 - a_2x_2 - b_2z_2 & \beta - a_1z_1 - b_1y_1 - a_2z_2 - b_2y_2 \\ -\beta - c_1x_1 - d_1z_1 - c_2x_2 - d_2z_2 & \xi - c_1z_1 - d_1y_1 - c_2z_2 - d_2y_2 \end{bmatrix}$$

The characteristic polynomial equation associated with A_{r_c} is

$$s^2 + m_1 s + m_2 = 0 \tag{5.45}$$

where

$$m_1 = a_1x_1 + d_1y_1 + (c_1 + b_1)z_1 + a_2x_2 + d_2y_2 + (b_2 + c_2)z_2 - \xi - \alpha \tag{5.46}$$

$$m_2 = [\ f_1y_1 + f_2y_2 + f_3z_2 + g_1\ ]\ x_1 + [\ f_4x_2 + f_5z_2 + g_2\ ]\ y_1 - f_1z_1^2 + [\ -f_3x_2 - f_5y_2 - (f_2 + f_4)z_2 + g_3\ ]\ z_1 + [\ f_6y_2 + g_4\ ]\ x_2 + g_5y_2 - f_6z_2^2 + g_6z_2 + g_7 \tag{5.47}$$

with $f_1,\ f_2,\ f_3,\ f_4,\ f_5,$ and f_6 defined as

$$f_1 = a_1d_1 - b_1c_1,\ f_2 = a_1d_2 - b_2c_1$$

$$f_3 = a_1c_2 - a_2c_1,\ f_4 = a_2d_1 - b_1c_2$$

$$f_5 = b_2d_1 - b_1d_2,\ f_6 = a_2d_2 - b_2c_2$$

and $g_1,\ g_2,\ g_3,\ g_4,\ g_5,\ g_6,$ and g_7 defined as

$$g_1 = c_1\beta - a_1\xi,\ \ g_2 = -b_1\beta - d_1\alpha$$

$$g_3 = -b_1\xi + (d_1 - a_1)\beta - c_1\alpha$$

$$g_4 = c_2\beta - a_2\xi,\ \ g_5 = -b_2\beta - d_2\alpha$$

$$g_6 = -b_2\xi + (d_2 - a_2)\beta - c_2\alpha$$

$$g_7 = \beta^2 + \alpha\xi$$

The characteristic polynomial equation associated with the closed-loop system which has the desired pole locations (i.e., $\gamma + \delta j, \phi - \delta j$) is

$$s^2 + n_1 s + n_2 = 0 \tag{5.48}$$

where

$$n_1 = \begin{cases} -2\gamma & \text{(complex conjugate case)} \\ -\gamma - \phi & \text{(real distinct case)} \end{cases} \tag{5.49}$$

$$n_2 = \begin{cases} \gamma^2 + \delta^2 & \text{(complex conjugate case)} \\ \gamma\phi & \text{(real distinct case)} \end{cases} \tag{5.50}$$

Since the requirement is to move the closed-loop poles to the specified locations, then Eq. (5.45) is equated to Eq. (5.48) to obtain

$$m_1 = n_1 \tag{5.51}$$

$$m_2 = n_2 \tag{5.52}$$

Equations (5.51) and (5.52) are the only equality constraint equation for this problem. Notice that Eqs. (5.51) and (5.52) represent an underdetermined system (i.e., there are two equations in six unknowns (x_1, y_1, z_1, x_2, y_2, and z_2)). Since the solution set of Eqs. (5.51) and (5.52) is not unique, we must impose additional optimization criteria for this problem. Again, notice that the inequality constraint equations for the optimization problem are $P_{i_r} > 0$, $(i = 1, 2)$. Possible candidates for additional optimization criteria in the case of pole placement for multiobjective control problems will be discussed in Chapter 6.

5.4 Recursive Algorithm

The multiple poles shifting problem using a game theoretic controller can be implemented by utilizing the following algorithm recursively

Step 1: Select a set of eigenvalue(s), Λ_i(a single pole or a pair of poles) of the current system matrix, A_i, to be shifted.

Step 2: Find the left eigenvector(s), v_i, of the current system matrix, A_i associated with the selected eigenvalue(s), Λ_i.

Step 3: Using the left eigenvetor(s), v_i, to form the transformation matrix, T_i.

Step 4: Form the reduced-set of system equation (Eq. (5.31) for a single pole shifting and Eq. (5.39) for a pair of poles).

Step 5: Using one of the superimposed optimization model proposed in the Chapter 6 to find optimal solutions, P_{1r} and P_{2r}

Step 6: Using Eqs. (5.26) and (5.30) to find solutions, P_{1i} and P_{2i}, to the full-set of the coupled algebraic Riccati equations, and the state feedback gain matrix, K_i.

Step 7: Form the new system matrix by

$$A_{i+1} = A_i - BK_i$$

Step 8: $i = i + 1$

Goto Step 1, if further pole shifting is necessary,

otherwise, stop.

For the multiple shifting case (i.e., recursively using the above algorithm), the proposed algorithm does not provide a guidance to choose the order for the shifting sequence. For a real control problem, the selection of shifting sequence may result in different optimum solutions. This phenomenon can be explained by the game theory with the Stackelberg strategy, since selecting the order of pole shifting in optimal pole placement problem is equivalent to selecting the order of importance for all optimization indices in game. Notice that for multiple shifting case, the open-loop poles can also be shifted simultaneously, which can avoid selection of the shifting sequence; however, the unknowns of the optimization model will be increased quadratically.

5.5 Summary

In this chapter, the principle of optimal pole placement in the game theoretic state feedback controller is provided and a recursive algorithm to implement pole placement task by utilization of game theoretic controller is provide. For the single pole shifting case, the solution of the LQR controller lies at a line segment, which is the solution set of the solution of the game theoretic controller. It infers that for a pair of poles shifting case, the solution set (i.e., a line segment) of the LQR controller lies in a 4-dimensional space, which is the solution set to the game theoretic controller. For both single pole and a pair of poles shifting cases, the solutions to the reduced system equations are not unique. Thus, a superimposed optimization criterion is need. Superimposed criterion can be chosen from system design specifications . The detail of those possible candidates will be introduced in Chapter 6.

Notice that in the optimal pole placement with game theoretic controller problems, the inequality constraints, $P_{i_r} > 0$, is a consequence of the "stabilizing" procedures (i.e., moving system open-loop poles from their current location to the left in s-plane). For the "de-stabilizing" task (i.e., moving the system open-loop poles from their current location to the right in s-plane), the inequality constraint can be released. Thus, for the later case, there is no inequality constraints imposed on P_{i_r}.

CHAPTER 6
SUPERIMPOSED OPTIMIZATION CRITERIA ON DESIGNING GAME THEORETIC STATE FEEDBACK CONTROLLERS

6.1 Introduction

In chapter 5, a new pole placement methodology for game theoretic state feedback controllers is proposed. There, it was shown that by using a game theoretic controller design, the design space could be expanded beyond that allowed by the LQR design. It was also shown that unlike the LQR approach with a single equation in one unknown (i.e., a single pole shifting case), an underdetermined system of equations resulted from the game theoretic approach. Thus, in order to determine an unique solution for the game theoretic controller, additional criteria must be superimposed. These criteria could be physical of artificial constraints. For example,

(1). Minimize the square of the Frobenius norm of the reduced-order state feedback gain matrix.

(2). Minimize the square of the Frobenius norm of the full-order state feedback gain matrix.

(3). Minimize the envelop bound of reduced-order state time response.

(4). Minimize the envelop bound of full-order state time response.

The Frobenius norm of the full (reduced) state feedback gain matrix is a measure of the magnitude of state feedback gain matrix, which is related to the control action on the full (reduced) model of the system. The envelop bound of the state time response for the full-order (reduced-order) system model restricts the possible state time response offset from the set point. The first three criteria are mathematically tractable, and are discussed in the following sections. The fourth criterion is numerically implementable, but is somewhat

intractable, due to the complexity in evaluating a matrix exponential. Beside these four proposed criteria, there exist other useful design specifications (e.g., the magnitude of weighted square displacements and velocities, and the magnitude of weighted square control effort [65], etc.). However, the optimization criteria on either "magnitude of weighted square displacements and velocities" or "magnitude of weighted square control effort" used in Reference 65. cannot be chosen as a superimposed optimization here, because each is merely a component of the performance indices (i.e., Eq. (2.36)). Thus, if either one of the above criterion is used, then it serves as a redundant criteria to the original multiobjective optimization problem.

The following describes the outline of this chapter. Section 6.2 discusses the optimization model based on the minimum square of the Frobenius norms of both the reduced-order and the full-order state feedback gain matrices. Section 6.3 discusses the minimum envelop bound of the reduced-order state time response, and Section 6.4 discusses the minimization of envelop bound of full-order state time response. Numerical examples to demonstrate the feasibility of the proposed schemes are presented throughout this chapter. The examples are based on an F-4 aircraft's lateral direction model. There is also another numerical example to demonstrate that the proposed methodology can easily be extended to the n-player differential game. This example is based a longitudinal motion model of an AIRC aircraft with three active controllers on board. This chapter is concluded with a summary in Section 6.5.

6.2 Minimize the Square of Frobenius Norm of Reduced-Order State Feedback Gain

In this section, the optimization model based on the minimization of the Frobenius norm of the reduced-order state feedback gain matrix, for both single-pole (in sub-section 6.2.1) and a pair of poles (in sub-section 6.2.2) shifting cases are discussed. The first-order necessary condition (or Kuhn-Tucker condition) and the second-order necessary condition will be derived for the both single-pole and a pair of poles shifting cases.

We observe that the Frobenius norm of the full-order state feedback gain matrix (K) and Frobenius norm of reduced-order state feedback gain matrix (K_r) are related as

$$\| K \|_F = \| K_r T^T \|_F \leq \| K_r \|_F \| T \|_F \tag{6.1}$$

Thus, the minimum of Frobenius norm of the full-order state feedback gain matrix is bounded by the product of the Frobenius norm of reduced-order state feedback matrix (K_r) and the Frobenius norm of the transformation matrix (T). Thus a discussion on the minimization of the Frobenius norm of the full-order state feedback gain matrix will not be presented.

6.2.1 Shifting a Single Pole

Given an $n \times n$ system equation (Eq. (2.1)), the single pole shifting problem with minimum square of the Frobenius norm of the reduced-order state feedback gain matrix can be treated as the following optimization problem (KR1):

$$\text{Minimize } (c_1 P_{1_r})^2 + (c_2 P_{2_r})^2$$

$$\text{subject to}$$

$$\mu = a - c_3 P_{1_r} - c_4 P_{2_r}$$

$$P_{1_r} > 0,\ P_{2_r} > 0. \tag{6.2}$$

where

$$c_1 = \| R_{11}^{-1} B_{1_r}^T \|_F \geq 0$$

$$c_2 = \| R_{22}^{-1} B_{2_r}^T \|_F \geq 0$$

$$c_3 = B_{1_r} R_{11}^{-1} B_{1_r}^T$$

$$c_4 = B_{1_r} R_{22}^{-1} B_{2_r}^T$$

The first-order and second-order necessary condition for the existence of a solution to the above optimization model are derived and analyzed below.

First-order Necessary Condition

Theorem 6.1 First-order Necessary Condition (Kuhn-Tucker Condition) [66]

Let x^* be a relative minimum of the following optimization problem (P1)

$$\text{Minimize } f(x)$$

$$\text{subject to}$$

$$h(x) = 0,\ g(x) \leq 0 \tag{6.3}$$

Supposing x^* is a regular point for the constraints (i.e., $\nabla h(x^*)$ and $\nabla g(x^*)$ are linear independent). Then there is a Lagrange multiplier, λ_1, for the equality constraint, and a Lagrange multiplier, λ_2, with $\lambda_2 \geq 0$ such that

$$\nabla f(x^*) + \lambda_1^T \nabla h(x^*) + \lambda_2^T \nabla g(x^*) = 0 \tag{6.4}$$

$$\lambda_2^T g(x^*) = 0 \tag{6.5}$$

❑

In Theorem 6.1, λ_2 may be non-zero only if its corresponding inequality constraint is active (i.e., the solution lies at the boundary). For the optimization model (KR1), the inequality constraints are strictly great than zero, therefore they are inactive. Based on this argument and Theorem 6.1, the first-order necessary conditions for the optimization model (KR1) are

$$2c_1 P_{1_r} + c_3 \lambda_1 = 0 \tag{6.6}$$

$$2c_2 P_{2_r} + c_4 \lambda_1 = 0 \tag{6.7}$$

where λ_1 is a scalar Lagrange multiplier for the equality constraint Eq. (5.31).

<u>Second-order Necessary and Sufficient Condition</u>

Theorem 6.2 Second-order Necessary and Sufficient Condition [66]

Suppose x^* is a regular point for the constraints (Eq. (6.3)). The sufficient (necessary) condition for x^* to be a strict relative minimum point of the optimization problem (P1) is

$$\nabla f(x^*) + \lambda_1^T \nabla h(x^*) + \lambda_2^T \nabla g(x^*) = 0 \tag{6.8}$$

$$\lambda_2^T g(x^*) = 0 \tag{6.9}$$

$$\lambda_2 \geq 0$$

and the Hessian matrix of the Lagrange equation,

$$L(x^*) = f(x^*) + \lambda_1^T h(x^*) + \lambda_2^T g(x^*)$$

is positive definite (semi-definite).

❑

For a single pole shifting problem, based the first-order necessary conditions (Eqs. (6.6) and (6.7)), the solutions of the reduced-order coupled algebraic Riccati equation (Eqs. (5.24) and (5.25)) are

$$P_{1_r} = -\frac{c_3\lambda_1}{2c_1},\ P_{2_r} = -\frac{c_4\lambda_1}{2c_2} \tag{6.10}$$

with

$$\lambda_1 = 2(\mu - \alpha)/(\frac{c_3^2}{c_1} + \frac{c_4^2}{c_2})$$

The Hessian matrix

$$\begin{bmatrix} 2c_1^2 & 0 \\ 0 & 2c_2^2 \end{bmatrix}$$

is positive semi-definite, since $c_1 \geq 0$ and $c_2 \geq 0$. Therefore, the solutions (Eq. (6.10)) are relative minimums for the optimization problem (KR1), since the solutions satisfy both the first-order and the second-order necessary conditions.

The non-linear optimization problem (KR1) is solved using "constr.m" in the MATLAB's Optimization Toolbox [67]. This MATLAB function is developed using a Sequential Quadratic Programming (SQP) method. In the SQP method, a Quadratic Programming (QP) subproblem is solved iteratively. The Hessian of Lagrangian is estimated and updated at each iteration using the BFGS formula [68,69].

Once the solution set, P_{1_r} and P_{2_r}, of the reduced-order coupled algebraic Riccati equation is obtained, the reduced-order state weighting matrices, Q_{1_r} and Q_{2_r}, can be obtained by substituting P_{1_r} and P_{2_r} into Eqs. (5.24) and (5.25). Finally, the solution of full-order coupled algebraic Riccati equation and full-order state weighting matrices can be obtained from Eqs. (5.26) and (5.22), respectively.

Recall from Theorem 5.1 in Section 5.3.2 that the solution of the LQR controller (Eq. (5.34)) belongs to the solution set of the game theoretic controller (Eq. (5.31)) for a single pole shifting case. Therefore, if the optimal solution to the optimization criteria on minimization for the Frobenius norm of the reduced-order state feedback gain matrix is equal to the solution to the LQR controller, then the optimal solution to the game theoretic controller is equal to the solution to the LQR controller; if the optimal solution to the above optimization criteria is the not equal to the solution to the LQR controller, then the optimal solution to the game theoretic controller is better than the solution to the LQR controller. Thus, in general, the solution of the above optimization model (KR1) is better than the solution to the LQR controller in the sense of minimization for the Frobenius norm of the reduced-order state feedback gain matrix.

Example 6.1:

Consider the same lateral direction model of the F-4 aircraft (Example 5.1) in Section 5.3.2. The controllable system is given as

$$\dot{x} = \begin{bmatrix} -1.7680 & 0.4125 & -14.2500 & 0.0000 \\ -0.0070 & -0.3831 & 6.0380 & 0.0000 \\ 0.0016 & -0.9975 & -0.1551 & 0.0586 \\ 1.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix} x + \begin{bmatrix} 1.7440 & 8.9520 \\ -2.9200 & -0.3075 \\ 0.0243 & -0.0036 \\ 0.0000 & 0.0000 \end{bmatrix} u$$

where as before

$$x = \begin{Bmatrix} p \\ r \\ \beta \\ \phi \end{Bmatrix} = \begin{Bmatrix} \text{roll rate} \\ \text{yaw rate} \\ \text{sideslip} \\ \text{roll angle} \end{Bmatrix} \quad \text{and} \quad u = \begin{Bmatrix} \delta_r \\ \delta_a \end{Bmatrix} = \begin{Bmatrix} \text{rudder} \\ \text{aileron} \end{Bmatrix}$$

The quadratic performance indices are

$$J_i = \int_0^{\infty} (x^T Q_i x + u_1^T R_{i1} u_1 + u_2^T R_{i2} u_2)\, dt \qquad (i = 1, 2)$$

The input weighting matrices are selected as follow

$$R_{11} = 2,\ R_{12} = 0,\ R_{21} = 0, \text{ and } R_{22} = 1$$

Partitioning the control task results in

$$B_1 = [1.7440 \quad -2.9200 \quad 0.0243 \quad 0.0000]^T$$

$$B_2 = [8.9520 \quad -0.3075 \quad -0.0036 \quad 0.0000]^T$$

To compare the game theoretic state feedback controllers with the LQR state feedback controller, the LQR performance index is chosen as

$$J = J_1 + J_2 = \int_0^{\infty} (x^T Q x + u^T R u)\, dt.$$

where the state weighting matrix, Q, and the control weighting matrix, R, are

$$Q = Q_1 + Q_2, \quad R = diag([R_{11} \ R_{22}])$$

The open-loop poles for the above system matrix of the F-4 aircraft lateral model are $[-0.0150,\ -0.2149 \pm 2.4858j,\ -1.8614]$. These poles correspond to the spiral mode, the Dutch roll mode, and the roll mode, respectively. Assume that the control task is to move the first open-loop pole of the system from its current location to a new location at -0.5 (i.e., to decrease the spiral mode time constant) and to minimize the Frobenius norm of the reduced-order state feedback gain matrix. Then, the solution to LQR controller (Eq. (5.34)) is

$$P_r = 0.0591$$

The corresponding Frobenius norm of the reduced-order state feedback gain matrix is 0.1623. Based on the superimposed optimization criterion that minimizes the Frobenius norm of the reduced-order state feedback gain matrix, the solutions to game theoretic controller (Eq. (5.31)) is

$$P_{1_r} = 0.1017,\ P_{2_r} = 0.0508$$

The corresponding Frobenius norm of the reduced-order state feedback gain matrix is 0.1570, which is less than the one obtained from the LQR design.

The solution set to game theoretic controller and the solution to LQR controller are graphically presented in Figure 6.1. Also shown in Figure 6.1 is the fact the LQR controller belongs to the solution set to game theoretic controller.

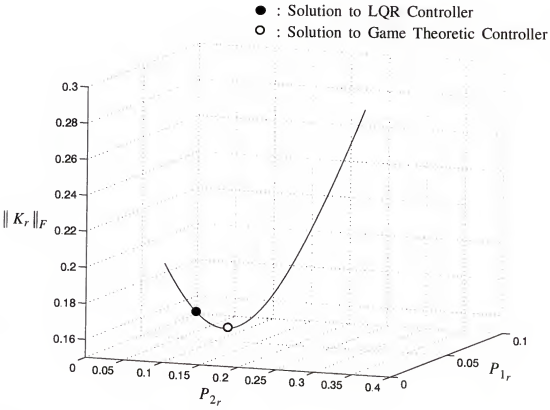

Fig. 6.1 The solution set (solid line) to game theoretic controller and the solution to LQR controller.

A time domain state response comparison between the two controllers is shown in Fig. 6.2. The initial conditions for the states are $x_0 = [0.1,\ 0.1,\ 0.1,\ 0.1]^T$. Figures 6.2(a), 6.2(b), 6.2(c), and 6.2(d) present correspondingly the roll rate, yaw rate, sideslip and roll angle state responses for the closed-loop system for both controller controllers. Figures 6.2(e) and 6.2(f) show the rudder and aileron responses. Under the present perturbed

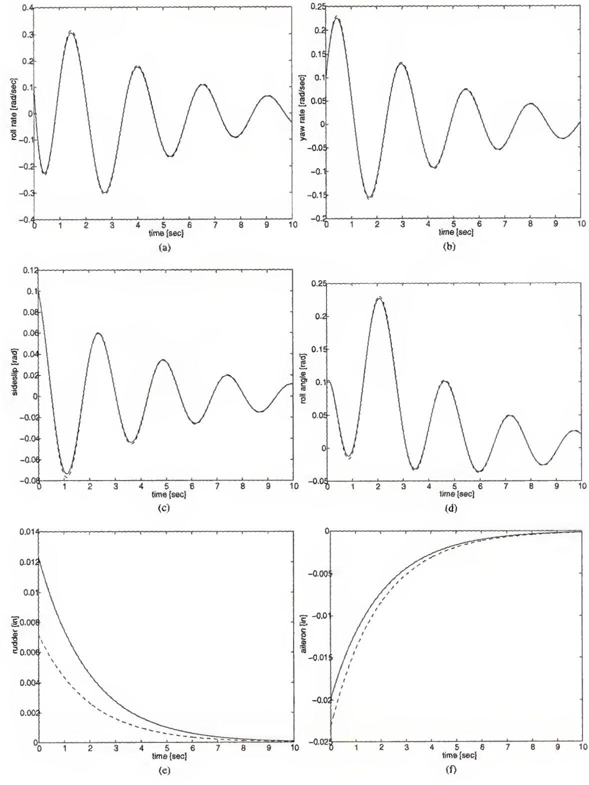

Figure 6.2 LQR controller vs. game theoretic controller (Legend: solid line: game theoretic controller (minimization of Frobenius norm of K_r), dash line: LQR controller (minimization of Frobenius norm of K_r).

configuration, the results show that the time responses of the roll rate, yaw rate, sideslip and roll angle states for the closed-loop system with the game theoretic controller are almost equal to their corresponding state responses for the closed-loop system with the LQR controller. The Frobenius norm of $u(t)$ ($= [u_1(t)\ \ u_2(t)]^T$) for the design based on the game theoretic controller are less than the one for LQR controller. For example, at $t = 0$, the Frobenius norm of the input for the game theoretic controller is 5.44×10^{-4} $(= 0.012^2 + (-0.02)^2)$; the Frobenius norm of the input for the LQR controller is 5.55×10^{-4} $(= 0.007^2 + (-0.0225)^2)$. The numerical result agrees with that the Frobenius norm of the reduced-order state feedback gain matrix using a game theoretic controller design is less than the one using a LQR controller design.

6.2.2 Shifting a Pair of Poles

Given an $n \times n$ system equation (Eq. (5.39)), a pair of poles shifting problem with minimum square of the Frobenius norm of the reduced-order state feedback gain matrix can be constructed using the following optimization problem (KR2):

$$\begin{gathered} \text{Minimize } \| K_r \|_F^2 \\ \text{subject to} \\ \text{Eqs. (5.51) and (5.52)} \\ P_{1_r} > 0, \;\; P_{2_r} > 0. \end{gathered} \tag{6.11}$$

where

$$K_r = \begin{bmatrix} R_{11}^{-1} B_{1_r}^T P_{1_r} \\ R_{22}^{-1} B_{2_r}^T P_{2_r} \end{bmatrix}$$

❑

First-order and Second-order Necessary Condition

The Lagrangian for the above optimization problem (KR2) is

$$L = \left\| \begin{bmatrix} R_{11}^{-1} B_{1_r} P_{1_r} \\ R_{22}^{-1} B_{2_r} P_{2_r} \end{bmatrix} \right\|_F + \lambda_1 (m_1 - n_1) + \lambda_2 (m_2 - n_2) \tag{6.12}$$

where m_i $(i = 1, 2)$ are defined in Eqs. (5.46) and (5.47), and n_i $(i = 1, 2)$ are defined in Eqs. (5.49) and (5.50). According to Theorem 6.1, the first-order necessary condition for the optimization problem (KR2) is

$$\nabla L(P_{1_r}^*, P_{2_r}^*) = 0$$

where $(P_{1_r}^*, P_{2_r}^*)$ is a relative minimum point of the optimization problem (KR2).

Based on Theorem 6.2, the second-order necessary (sufficient) condition for the optimization problem (KR2) is that the Hessian matrix of the Lagrangian (Eq. (6.12)) is positive semi-definite (definite).

A MATLAB code in the Optimization Toolbox, "constr.m," is used to solve the non-linear optimization problem (KR2). Again, once the solution set, P_{1_r} and P_{2_r}, of the reduced-order coupled algebraic Riccati equation is obtained, the reduced-order state weighting matrices Q_{1_r} and Q_{2_r} can be obtained by substituting P_{1_r} and P_{2_r} into Eqs. (5.24) and (5.25), then, the solution of full-order coupled algebraic Riccati equation and full-order state weighting matrices can be obtained from Eqs. (5.26) and (5.22) respectively.

Example 6.2:

Consider the same lateral motion model of the F-4 aircraft from Example 6.1. The open-loop system matrix A and the control influence matrices B, B_1 and B_2 have the same value as they did in Example 6.1. The input weighting matrices are given as $R_{11} = 1$, $R_{12} = 0$, $R_{21} = 0$, $R_{22} = 1$, and $R = diag([1 \;\; 1])$.

The open-loop poles for the system matrix of the F-4 aircraft lateral model from Example 6.1 are $[-0.0150, \; -0.2149 \pm 2.4858j, \; -1.8614]$. In the current example, the control task is to move the pair of complex conjugate poles of the system from

their current location to a new location at $-1 \pm 2j$. This is equivalent to adding more damping to the Dutch roll mode.

Numerical results for the full state feedback gain matrix and closed-loop system matrix are provided in Table 6.1. In Tables 6.1, the "game theoretic controller" represents the state feedback controller based a differential game design, and the "LQR controller" represents the state feedback controller based on the linear quadratic regulator design.

In Table 6.1, the results show that the game theoretic controller has a smaller $\| K_r \|_F$ than the LQR controller. Hence, if the state time responses for both closed-loop systems are very close, then the square of the total input response of the closed-loop system using the game theoretic controller should be smaller than those for the closed-loop system using the LQR controller.

Time domain state response comparisons between the two controllers are shown in Fig. 6.3. The initial conditions for the states are $x_0 = [0.1,\ 0.1,\ 0.1,\ 0.1]^T$. Figures 6.3(a), 6.3(b), 6.3(c), and 6.3(d) present correspondingly the roll rate, yaw rate, sideslip and roll angle state responses for the closed-loop systems using the game theoretic controller and the LQR controller. Figures 6.3(e) and 6.3(f) present the time response of both the rudder and aileron inputs for both closed-loop systems. The graphical results show that the time responses for the roll rate state, yaw rate state, sideslip state and rudder input are about the same for both closed-loop systems.

However, the overshoot percentages of time responses of the roll angle states and aileron input for the closed-loop system utilizing the game theoretic controller is less than those for the closed-loop system using the LQR controller. These results are again in agreement with that the Frobenius norm of the reduced-order state feedback gain matrix obtained by a game theoretic controller design being less than the one obtained by a LQR controller design.

Table 6.1 Resulting Gains and Closed-loop Matrices

Game Theoretic Controller

K	$\begin{bmatrix} 0.0095 & -0.5239 & -0.4232 & 0.0321 \\ 0.0017 & -0.0609 & -0.1215 & 0.0040 \end{bmatrix}$
A_{c_1}	$\begin{bmatrix} -1.7995 & 1.8718 & -12.4262 & -0.0919 \\ 0.0212 & -1.9317 & 4.7650 & 0.0951 \\ 0.0014 & -0.9850 & -0.1453 & 0.0578 \\ 1.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix}$
$\Vert K_r \Vert_F$	1.5272

LQR Controller

K	$\begin{bmatrix} 0.0095 & -0.5208 & -0.4265 & 0.0320 \\ 0.0019 & -0.0842 & -0.1176 & 0.0053 \end{bmatrix}$
A_{c_1}	$\begin{bmatrix} -1.8017 & 2.0745 & -12.4539 & -0.1036 \\ 0.0212 & -1.9296 & 4.7566 & 0.0950 \\ 0.0014 & -0.9851 & -0.1452 & 0.0578 \\ 1.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix}$
$\Vert K_r \Vert_F$	1.5285

6.3 Minimize the Envelop Bound of the Reduced-Order State Response

In this section, the optimization model based on the minimal envelop bound of the reduced-order state response will be constructed for a pair of poles shifting case. The following derivation will not be applied to a single pole shifting case, since the time response of the reduced-order, 1-by-1 system, with a prescribed eigenvalue, is fixed.

In order to facilitate future derivations, the matrix exponential of a 2×2 system is first reviewed.

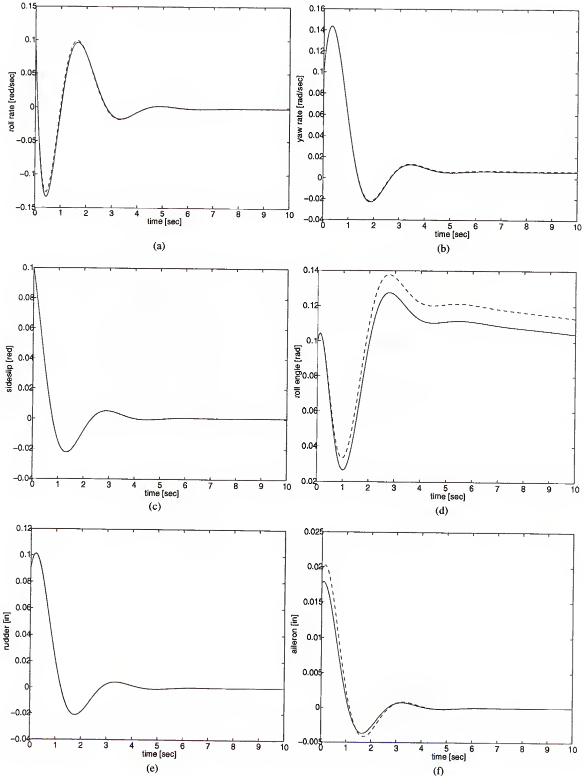

Figure 6.3 LQR controller vs. game theoretic controller (Legend: solid line: game theoretic controller (minimization of Frobenius norm of K_r), dash line: LQR controller (minimization of Frobenius norm of K_r).

Theorem 6.3 The exponential of a general 2×2 matrix (Bernstein and So [70])

Let λ and μ denote the eigenvalues of $A \in \mathbb{C}^{2\times 2}$, then

$$e^{At} = e^{\lambda t}[(1 - \lambda)I + A], \qquad \text{if } \mu = \lambda \tag{6.13}$$

$$e^{At} = \frac{\mu e^{\lambda t} - \lambda e^{\mu t}}{\mu - \lambda} I + \frac{e^{\lambda t} - e^{\mu t}}{\mu - \lambda} A, \qquad \text{if } \mu \neq \lambda \tag{6.14}$$

Proof: Refer to Reference 70.

Theorem 6.3 and Eq. (5.39) infer that Frobenius norm of envelop of the reduced state response, $x_r(t)$, is bounded by Frobenius norm of A_{r_c}. Hence the optimization model (AR2) is defined as follow

$$\begin{gathered} \text{Minimize } \| A_{r_c} \|_F \\ \text{subject to} \\ \text{Eqs. (5.51) and (5.52)} \\ P_{1_r} > 0, \ P_{2_r} > 0. \end{gathered} \tag{6.15}$$

where A_{r_c} is defined Eq. (5.39).

□

<u>First-order and Second-order Necessary Condition</u>

The Lagrangian for the above optimization problem (AR2) is

$$L = \| A_{r_c} \|_F + \lambda_1(m_1 - n_1) + \lambda_2(m_2 - n_2) \tag{6.16}$$

where A_{r_c} is defined in Eq. (5.39), m_i $(i = 1, 2)$ are defined in Eqs. (5.46), (5.47), and n_i $(i = 1, 2)$ are defined in Eqs. (5.49), (5.50). According to Theorem 6.1, the first-order necessary condition for the optimization problem (AR2) is

$$\nabla L(P_{1_r}^*, P_{2_r}^*) = 0$$

where $(P_{1_r}^*, P_{2_r}^*)$ is a relative minimum point of the optimization problem (AR2).

Based on Theorem 6.2, the second-order necessary (sufficient) condition for the optimization problem (AR2) is that the Hessian matrix of the Lagrangian (Eq. (6.16)) is positive semi-definite (definite).

The solution set P_{1_r} and P_{2_r} is again obtained using "constr.m" in MATLAB's Optimization Toolbox. The reduced-order state weighting matrices, Q_{1_r} and Q_{2_r}, can be obtained by substituting P_{1_r} and P_{2_r} into Eqs. (5.24), (5.25). The solution of full-order coupled algebraic Riccati equation and full-order state weighting matrices can be obtained from Eqs. (5.26) and (5.22), respectively.

Example 6.3:

Consider the same lateral direction model of the F-4 aircraft from Example 6.2. The open-loop system matrix A, the control influence matrices B, B_1 and B_2 and the input weighting matrices R, R_{11}, $R_{12,}$ R_{21} and R_{22} have the same values as shown in Example 6.2.

The open-loop poles for the system matrix of the lateral motion model of an F-4 aircraft are located at $[-0.0150,\ -0.2149 \pm 2.4858j,\ -1.8614]$. The controller design task is: (1) move a pair of complex conjugate open-loop poles from their current location to the new location at $[-1,\ -1.5]$, (2) move two real open-loop poles from their current location to their new location at $-1 \pm 2j$, (3) minimize the envelop bound of the reduced-order state response. The reason for moving the complex poles to real locations and vice versa is merely to demonstrate the methodology. The numerical results are shown in Tables 6.2 and 6.3. In Table 6.2, a pair of complex conjugate poles of the original system matrix are shifted to the location at $[-1,\ -1.5]$. The full state feedback gain matrix (K_1) and the intermediated closed-looped system matrix, A_{c_1}, along with its associated Frobenius norm of the reduced-order system matrix A_{r_c}, are tabulated in Table 6.2. Matrix A_{c_1} has the two new real poles located at $[-1,\ -1.5]$ and the two original real poles of the original open-loop system matrix remains unchanged.

Table 6.2 Gains and Closed-loop Matrices (Complex Conjugate Poles to Two Real Poles)

Game Theoretic Controller

K_1 $$\begin{bmatrix} 0.0030 & -0.7008 & -0.6545 & 0.0385 \\ 0.1150 & 3.5684 & -19.6990 & -0.1365 \end{bmatrix}$$

A_{c_1} $$\begin{bmatrix} -2.8023 & -30.3097 & 160.9536 & 1.1545 \\ 0.0370 & -1.3322 & 1.8918 & 0.0706 \\ 0.0019 & -0.9676 & -0.2419 & 0.0572 \\ 1.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix}$$

$\| A_{rc_1} \|_F$ 1.8577

LQR Controller

K_1 $$\begin{bmatrix} 0.0054 & -0.3383 & -0.1801 & 0.0204 \\ 0.1113 & -0.1012 & -13.8549 & 0.0565 \end{bmatrix}$$

A_{c_1} $$\begin{bmatrix} -2.7737 & 1.9081 & 110.0923 & -0.5414 \\ 0.0429 & -1.4021 & 1.2516 & 0.0770 \\ 0.0019 & -0.9896 & -0.2006 & 0.0583 \\ 1.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix}$$

$\| A_{rc_1} \|_F$ 2.4707

In Table 6.3, the two original real poles of the intermediated closed-loop system matrix are shifted from their current locations (i.e., -0.0150 and -1.8614) to their new locations (i.e., $-1 \pm 2j$). The full state feedback gain matrix (K_2), and the final closed-loop system matrix, A_{c_2}, with its associated Frobenius norm of the reduced-order system matrix A_{r_c} are also listed in Table 6.3. The final closed-loop system matrix, A_{c_2}, has a pair of complex conjugate poles located at $-1 \pm 2j$, while the two real poles (i.e., $[-1, -1.5]$) of the intermediated system matrix remains unchanged.

Table 6.3 Gains and Closed-loop Matrices (Two Real Poles to Complex Conjugate Poles)

Game Theoretic Controller

K_2	$\begin{bmatrix} -0.0051 & 0.0033 & 0.9476 & 0.0032 \\ -0.0164 & -0.9369 & 6.3016 & 0.0869 \end{bmatrix}$
A_{c_2}	$\begin{bmatrix} -2.6470 & -21.9280 & 102.8892 & 0.3710 \\ 0.0172 & -1.6107 & 6.5965 & 0.1067 \\ 0.0020 & -0.9711 & -0.2423 & 0.0574 \\ 1.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix}$
$\lVert A_{rc_2} \rVert_F$	3.2663

LQR Controller

K_2	$\begin{bmatrix} -0.0050 & 0.0048 & 2.2237 & 0.0215 \\ 0.085 & -0.0081 & -3.8011 & -0.0368 \end{bmatrix}$
A_{c_2}	$\begin{bmatrix} -2.8409 & 1.9726 & 140.2429 & -0.2494 \\ 0.0311 & -1.3907 & 6.5760 & -0.1285 \\ 0.0020 & -0.9898 & -0.2683 & 0.0577 \\ 1.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix}$
$\lVert A_{rc_2} \rVert_F$	4.1769

In Tables 6.2 and 6.3, the "Game Theoretic Controller" represents the state feedback controller based on a differential game design, and the "LQR Controller" represents the state feedback controller based on a linear quadratic regulator design. Also in Table 6.2, the subscript "1" represents the first shifting step and in Table 6.3, the subscript "2" represents the second shifting step.

The numerical results demonstrate that the game theoretic controller has a smaller $\lVert A_{rc} \rVert_F$, $i = 1, 2$, than the LQR controller. Therefore, the state responses for the

closed-loop system using the game theoretic controller should have tighter response envelops as compared with the LQR design.

The initial conditions for the states are $x_0 = [0.1,\ 0.1,\ 0.1,\ 0.1]^T$. The time responses of the roll rate, yaw rate, sideslip and roll angle states of the closed-loop systems for both controllers are shown in Figures 6.4(a), 6.4(b), 6.4(c), and 6.4(d), respectively. The rudder and aileron state time responses are shown in Figures 6.4(e) and 6.4(f), correspondingly.

These results demonstrate that the maximum overshoots of time responses associated with the roll rate, yaw rate, sideslip and roll angle states for the closed-loop system using the game theoretic controller are less than those maximum overshoots associated with corresponding state responses for the closed-loop system using the LQR controller. The results satisfy the prediction that the closed-loop system using a game theoretic controller has a narrower envelop of the state time response envelop than the one using a LQR controller.

6.4 Minimize the Envelop Bound of the Full-Order State Response

The state response of the full-order closed-loop system is

$$x(t) = e^{A_c t} x_0 \tag{6.17}$$

where A_c is the full-order closed-loop system matrix

$$A_c = A - \sum_{i=1}^{2} B_i R_{ii}^{-1} B_{i_r}^T P_{i_r} T^T \tag{6.18}$$

The 2-norm of the state response, $x(t)$, can be expressed as

$$\begin{aligned} \|x(t)\|_2 &= \|e^{A_c t} x_0\|_2 \\ &\leq \|e^{A_c t}\|_2 \ \|x_0\|_2 \end{aligned} \tag{6.19}$$

The bound of the 2-norm of the state response depends only on the 2-norm of matrix exponential $e^{A_c t}$, since the 2-norm of x_0 is fixed for a give perturbed configuration of the

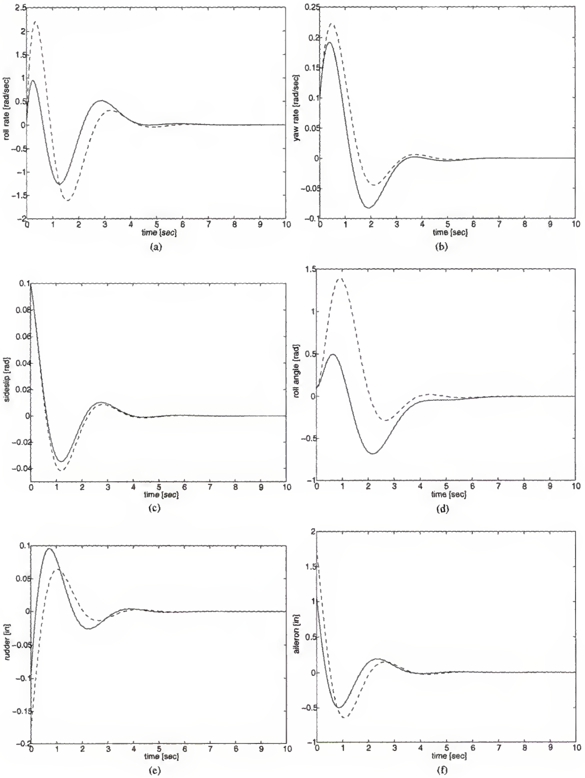

Figure 6.4 LQR controller vs. game theoretic controller (Legend: solid line: game theoretic controller (minimization of Frobenius norm of A_{rc}), dash line: LQR controller (minimization of Frobenius norm of A_{rc}).

linear system. The following theorem is used to determine the 2-norm of a matrix exponential.

Theorem 6.4 (Van Loan [63])

Given a constant matrix $A \in \mathbb{C}^{n\times n}$, the Schur decomposition states that there exists a unitary matrix $Q \in \mathbb{C}^{n\times n}$, such that

$$Q^H A Q = D + N \tag{6.20}$$

where D is a diagonal matrix and N is a strictly upper triangular matrix. The diagonal entries of D consist of system eigenvalues located at the diagonal entries. The 2-norm of e^{At} is bounded by

$$\left\|e^{A\,t}\right\|_2 \leq e^{\alpha(A)\,t} \sum_{k=0}^{n-1} \frac{\|Nt\|_2^k}{k!} \tag{6.21}$$

where

$$\alpha(A) = \max\{\,\mathrm{Re}(\lambda) \mid \lambda \in \lambda(A)\}. \tag{6.22}$$

Proof: The poof of this theorem is available in the Reference 63.

Based on theorem 6.4, the bound of the 2-norm of the matrix exponential $e^{A_c t}$ is determined by the 2-norm of N, since $\alpha(A)$ and t are fixed for a linear system with prescribed eigenvalues at a specified time t.

The above optimization criterion (i.e., the 2-norm of N) can be superimposed on the constrain equation (Eq. (5.31) for a single pole shifting case, or Eqs. (5.51), (5.52) for a pair of poles shifting case) to form an optimization model (AUC) described in the following section for "unconstrained" full-order state feedback gain case. Notice that the term, "unconstrained," means that there are no constraints imposed on the numerical value of each individual entry in the state feedback gain matrix.

Unconstrained Full-Order State Feedback Gain Case:

According to theorem 6.4 and Eqs. (6.19), (6.21), the 2-norm of the state response, $x(t)$, is bounded by 2-norm of N. Hence, the optimization criterion is chosen to minimize the 2-norm of N to obtain a tighter bound of full state response envelop. Based on that criterion, the optimization problem (AUC) can be stated as

$$\text{Minimize } \|N\|_2$$

$$\text{subject to}$$

$$\mu = a - B_{1_r} R_{11}^{-1} B_{1_r}^T P_{1_r} - B_{2_r} R_{22}^{-1} B_{2_r}^T P_{2_r}$$

(or Eqs. (5.51) and (5.52) for shifting a pair of poles)

$$P_{1_r} > 0, \; P_{2_r} > 0.$$

$\square$

First-order and Second-order Necessary Condition

The Lagrangian for the above optimization problem (AUC) is

$$L = \begin{cases} \| N \|_2 + \lambda_1(a - \mu B_{1_r} R_{11}^{-1} B_{1_r}^T P_{1_r} - B_{2_r} R_{22}^{-1} B_{2_r}^T P_{2_r}) & \text{(single pole)} \\ \| N \|_2 + \lambda_1(m_1 - n_1) + \lambda_2(m_2 - n_2) & \text{(a pair of poles)} \end{cases} \tag{6.23}$$

According to Theorem 6.1, the first-order necessary condition for the optimization problem (AUC) is

$$\nabla L(P_{1_r}^*, P_{2_r}^*) = 0$$

where $(P_{1_r}^*, P_{2_r}^*)$ is a relative minimum point of the optimization problem (AUC).

Based on Theorem 6.2, the second-order necessary (sufficient) condition for the optimization problem (AUC) is that the Hessian matrix of the Lagrangian (Eq. (6.23)) is positive semi-definite (definite).

The solution set, P_{1_r} and P_{2_r}, of the non-linear optimization problem (AUC) is obtained using "constr.m" in MATLAB's Optimization Toolbox. The reduced-order state weighting matrices, Q_{1_r} and Q_{2_r}, can be obtained by substituting P_{1_r} and P_{2_r} into Eqs.

(5.24), (5.25). Finally, the solution of full-order coupled algebraic Riccati equations and full-order state weighting matrices can be obtained from Eqs. (5.26), (5.22), respectively.

Recall from Theorem 5.1 in Section 5.3.2 that the solution of the LQR controller (Eq. (5.34)) belongs to the solution set of the game theoretic controller (Eq. (5.31)) for a single pole shifting case. Therefore, if the optimal solution to the above optimization criterion is not equal to the solution to the LQR controller, then the optimal solution to the game theoretic controller must be better than the solution to the LQR controller. Thus, in general the solution of the above optimization model (AUC) (i.e., the optimal solution to the game theoretic controller) is better than the solution to the LQR controller in the sense of minimization of the 2-norm of the state response to the full-order closed-loop matrix.

Example 6.4: the Unconstrained Case

Consider the same lateral direction model of the F-4 aircraft from Example 6.2. The open-loop system matrix (or the state influence matrix) A, the control influence matrices $B,\ B_1,\ B_2$ and the input weighting matrices $R,\ R_{11},\ R_{12},\ R_{21},\ R_{22}$ all have the same values as they did in Example 6.2.

The open-loop poles for the system matrix of the lateral motion model of the F-4 aircraft are located at $[-0.0150,\ -0.2149 \pm 2.4858j,\ -1.8614]$. The control task is designed to move the open-loop poles from current location to $[-0.5,\ -1 \pm 2j,\ -2]$ (i.e., decrease the time constants of all the modes and add damping to the Dutch roll mode), while minimize the envelop bound of the full-order state response.

The numerical results are shown in Tables 6.4 and 6.5. In Table 6.4, a pair of complex conjugate poles of the original system matrix are shifted from their current location to $-1 \pm 2j$. The full state feedback gain matrix (K_1) and the intermediated closed-looped system matrix , A_{c_1}, along with its associated 2-norm of N are tabulated in Table 6.4. Matrix A_{c_1} has a pair of complex conjugate poles located at $-1 \pm 2j$, while the two real poles of the original open-loop system matrix remain unchanged.

Table 6.4 Resulting Gains and Closed-loop Matrices (Complex Conjugate Poles)

	Game Theoretic Controller
K_1	$\begin{bmatrix} 0.0080 & -0.4979 & -0.2790 & 0.0301 \\ 0.0122 & 0.0183 & -1.5627 & 0.0046 \end{bmatrix}$
A_{c_1}	$\begin{bmatrix} -1.8913 & 1.1173 & 0.2262 & -0.0940 \\ 0.0202 & -1.8312 & 4.7428 & 0.0893 \\ 0.0014 & -0.9853 & -0.1539 & 0.0579 \\ 1.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix}$
$\lVert N_1 \rVert_2$	4.1339
	LQR Controller
K_1	$\begin{bmatrix} 0.0079 & -0.4837 & -0.2899 & 0.0293 \\ 0.0135 & -0.0807 & -1.5748 & 0.0105 \end{bmatrix}$
A_{c_1}	$\begin{bmatrix} -1.9023 & 1.9787 & 0.3531 & -0.1448 \\ 0.0203 & -1.8204 & 4.7073 & 0.0888 \\ 0.0015 & -0.9860 & -0.1537 & 0.0579 \\ 1.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix}$
$\lVert N_1 \rVert_2$	4.1615

In Table 6.5, the two real poles of the intermediated closed-loop system matrix are shifted from their current location (i.e., $[-0.0150 \;\; -1.8614]$) to the new location (i.e., $[-0.5 \;\; -2]$), respectively. The full state feedback gain matrix K_2, and the final closed-loop system matrix A_{c_2} (with its associated 2-norm of N) are also listed in Table 6.5. The final closed-loop system matrix, A_{c_2}, has the real poles located at -0.5 and -2, while the pair of complex conjugate poles of the intermediated system matrix remain unchanged.

Table 6.5 Resulting Gains and Closed-loop Matrices (Two Real Poles)

Game Theoretic Controller

K_2

$$\begin{bmatrix} -0.3384 & 0.0046 & -0.3923 & -0.6345 \\ 0.1380 & -0.0082 & 0.1594 & 0.2320 \end{bmatrix}$$

A_{c_2}

$$\begin{bmatrix} -2.5360 & 1.1830 & -0.5161 & -1.0640 \\ -0.9256 & -1.8202 & 3.6462 & -1.6922 \\ 0.0102 & -0.9855 & -0.1438 & 0.0741 \\ 1.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix}$$

$\|N_2\|_2$ 3.4656

LQR Controller

K_2

$$\begin{bmatrix} -0.0198 & -0.1583 & -0.0526 & -0.4113 \\ 0.0241 & 0.0587 & 0.0536 & 0.1847 \end{bmatrix}$$

A_{c_2}

$$\begin{bmatrix} -2.0833 & 1.7293 & -0.0347 & -1.0807 \\ -0.0300 & -2.2644 & 4.5702 & -1.0553 \\ 0.0020 & -0.9820 & -0.1523 & 0.0686 \\ 1.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix}$$

$\|N_2\|_2$ 4.2328

In Table 6.4, the subscript “1” represents the first shifting step and in Table 6.5, the subscript “2” represents the second shifting step. The results show the game theoretic controller has a smaller $\|N_i\|_2\,(i = 1, 2)$ than the LQR controller. Hence, the state response for the closed-loop system using the game theoretic controller should result in a tighter response envelop compared to the LQR design.

Figure 6.5 shows the state responses for both controllers. The initial conditions for the states are $[0,\ 0.1,\ 0.1,\ 0]^T$. Figures 6.5(a)-(d) represent, respectively, the roll rate, yaw rate, sideslip and roll angle state responses for the closed-loop system (after two successive shifts) for both controllers. The input responses are shown in Figures 6.5(e) and 6.5(f).

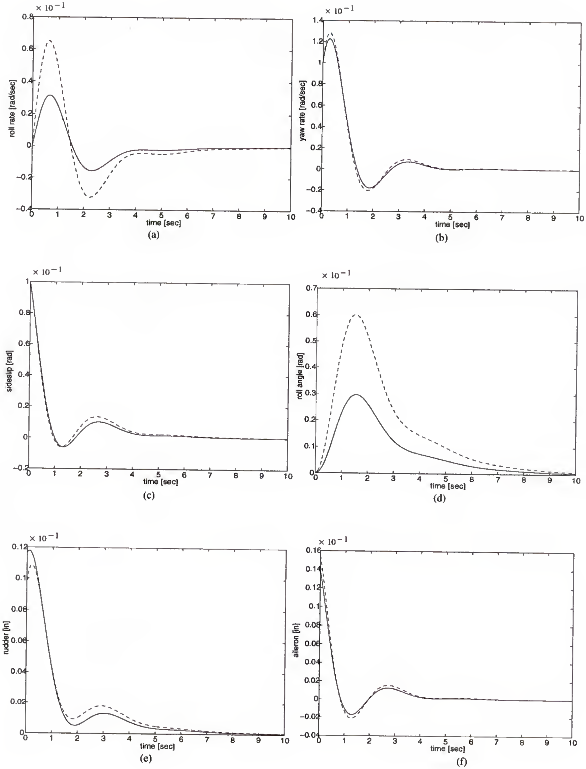

Figure 6.5 LQR controller vs. game theoretic controller (Legend: solid line: game theoretic controller (minimization of full-order state response), dash line: LQR controller (minimization of full-order state response).

Based on the current perturbed configuration, the results show that the time responses for all the sideslip state, yaw rate state, rudder input, and aileron input are about the same for both closed-loop systems. However, those overshoots associated with both the roll rate and roll angle state responses from the game theoretic controller are approximately 50% of the overshoots of their associated state responses generated from the LQR controller. These results are in agreement with the $\| N_2 \|_2$ results in Table 6.5.

Up to this point, all the proposed optimization models only implement pole shifting tasks. However, as mentioned in chapter 5, it is possible for system designers to add other physical or "artificial" design specification into the above optimization models. These details are discuss in next sub-section.

Constrained Full-Order State Feedback Gain Case:

Consider a practical design situation where the control system engineer wants to modify an existing state feedback system in the following manner: The engineer is not satisfied with the time response of the resulted system. He wants to relocate a single pole of the system without changing all the entries of the full-order state feedback gain matrix due to design considerations (e.g., less modification effort, or physical constraints on the state feedback gain matrix). Thus, the task is to move the original open-loop pole to new location, while keeping some elements of state feedback gain matrix fixed.

Based on this assumption, the constrained optimization problem (AC) can be stated as

Minimize $\|N\|_2$

subject to

$$\mu = \alpha - B_{1_r} R_{11}^{-1} B_{1_r}^T P_{1_r} - B_{2_r} R_{22}^{-1} B_{2_r}^T P_{2_r}$$

(or Eqs. (5.51) and (5.52) for shifting of a pair of poles)

$$P_{1_r} > 0,\ P_{2_r} > 0,\ K_{ij} = \left\{ \begin{bmatrix} R_{11}^{-1} B_{1_r}^T P_{1_r} \\ R_{22}^{-1} B_{2_r}^T P_{2_r} \end{bmatrix} T^T \right\}_{ij} = c_{ij}$$

The quantity c_{ij} in the above model is a constant, and the subscript "*ij*" represents the constrained entry located at *i*-th row and *j*-th column, in the state feedback gain matrix (K).

Notice that this practical design problem cannot be implemented by conventional LQR pole placement technique, since the LQR pole placement methodology yields an unique solution as shown in Section 5.3.2.

Example 6.5: the Constrained State Feedback Gain Case

In this example, the design task does not only perform the single pole shifting, but also imposes a state feedback gain constraint. The open-loop poles for the above system matrix of an F-4 aircraft lateral model are $[-0.0150,\ -0.2149 \pm 2.4858j,\ -1.8614]$. In this example, it is assumed that there exists a designed state feedback controller which moved the first open-loop pole of the system from its open-loop location to a location at -0.5 (Case 1 in Table 6.6) and the associated state feedback gain matrix is determined.

However, the system performance is not satisfactory and it is desired to move the first pole to a new location at -1, while keeping the second row of the obtained state feedback gain matrix fixed (Case 2 in Table 6.6). The reason for keeping the second row of state feedback gain matrix unchanged may be due to a geometric constraint on the aileron angle.

The numerical results of the full feedback gain matrix (K), the closed-looped system matrix, A_c, along with its associated 2-norm of N are tabulated in Table 6.7. In Table 6.7, the subscripts "1" and "2" represent "Case 1" and "Case 2," respectively, from Table 6.6. The constraint of the optimization model for the game theoretic controller of Case 2 is fulfilled as shown in the Table 6.7 (i.e., the second row of K_2 is the same as the one of K_1).

The graphical comparison between the two controllers are shown in Fig. 6.6. The initial conditions for the states are $x_0 = [0,\ 0.1,\ 0.1,\ 0]^T$. Figures 6.6(a), 6.6(b), 6.6(c), and 6.6(d) present the time responses for the roll rate, the yaw rate, the sideslip and the roll angle states of the closed-loop system for both controllers.

Table 6.6 Design Specification for State Feedback Controllers in Example 6.5

Case	Method	Closed-loop Pole Location	Constraints on State Feedback Gain
1	Nash	$[-0.5, -0.2149 \pm 2.4858j, -1.8614]$	No Constraints
2	Nash	$[-1.0, -0.2149 \pm 2.4858j, -1.8614]$	State Feedback Gain (Aileron) Same as Case 1

The graphical results show that the overshoots of the time state response associated with Case 1 (in Table 6.6) is smaller than the ones with Case 2 (in Table 6.6). It agrees with the prediction based on the 2-norm of N. The rudder input for Case 2 is greater than the one for Case 1, and the aileron input for the Case 2 is smaller than the one for Case 1. This phenomena is due to the constraint on the state feedback gain matrix for Case 2.

Extension to the N-Player Differential Games

Previous numerical examples (i.e., Examples 6.1 to 6.5) has demonstrated the capability of shifting either a single pole or a pair of poles from their current location(s) to any location in the s-plane for the 2-player cases. As mentioned in Chapter 5, the proposed methodology can be easily extend to solve an n-player differential game. The following example will be used to the demonstrate this extensity.

Example 6.6:

Consider a longitudinal motion model of an AIRC aircraft [71]. The controllable system is given as

$$\dot{x} = Ax + Bu \tag{6.24}$$

$$A = \begin{bmatrix} 0.0000 & 0.0000 & 1.1320 & 0.0000 & -1.0000 \\ 0.0000 & -0.0538 & -0.1712 & 0.0000 & 0.0705 \\ 0.0000 & 0.0000 & 0.0000 & 1.0000 & 0.0000 \\ 0.0000 & 0.0485 & 0.0000 & -0.8556 & -1.0130 \\ 0.0000 & -0.2909 & 0.0000 & 1.0532 & -0.6859 \end{bmatrix}$$

Table 6.7 Resulting Gains and Closed-loop Matrices

Game Theoretic Controller (1st pole from –0.015 to –0.5)

K_1

$$\begin{bmatrix} 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0589 & 0.1384 & -0.0267 & 0.1043 \end{bmatrix}$$

A_{c_1}

$$\begin{bmatrix} -2.2955 & -0.8268 & -14.0108 & -0.9337 \\ 0.0111 & -0.3405 & 6.0298 & 0.0321 \\ 0.0018 & -0.9970 & -0.1552 & 0.0590 \\ 1.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix}$$

$\|N_1\|_2$ 14.8640

Game Theoretic Controller (1st pole from –0.015 to –1 and aileron feedback gain fixed)

K_2

$$\begin{bmatrix} -0.0975 & -0.2292 & 0.0442 & -0.1727 \\ 0.0589 & 0.1384 & -0.0267 & 0.1043 \end{bmatrix}$$

A_{c_2}

$$\begin{bmatrix} -2.1252 & -0.4267 & -14.0880 & -0.6323 \\ -0.2737 & -1.0097 & 6.1589 & -0.4721 \\ 0.0042 & -0.9914 & -0.1563 & 0.0632 \\ 1.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix}$$

$\|N_2\|_2$ 14.7826

$$B = \begin{bmatrix} 0.0000 & 0.0000 & 0.0000 \\ -0.1200 & 1.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 \\ 4.4190 & 0.0000 & -1.6650 \\ 1.5750 & 0.0000 & -0.0732 \end{bmatrix}$$

The state and input variables are defined as

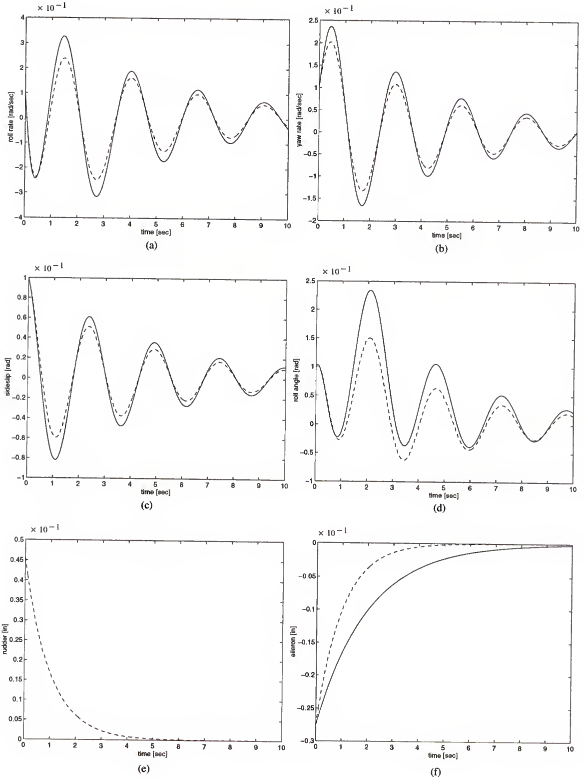

Figure 6.6 Legend: Solid line: game theoretic controller (1st pole from –0.015 to –0.5), dashed line: game theoretic controller (1st pole from –0.015 to –1 with 2nd row of K_2 given)

$$x = \begin{Bmatrix} Z \\ u \\ \theta \\ q \\ w \end{Bmatrix} = \begin{Bmatrix} \text{altitude [m]} \\ \text{forward speed [m/sec]} \\ \text{pitch angle [degrees]} \\ \text{pictch rate [deg/sec]} \\ \text{vertical speed [m/sec]} \end{Bmatrix}$$

$$u = \begin{Bmatrix} \delta_s \\ \dot{u} \\ \delta_e \end{Bmatrix} = \begin{Bmatrix} \text{spoiler angle [degrees/10]} \\ \text{forward acceleration [m/sec}^2\text{]} \\ \text{elevator angle [degrees]} \end{Bmatrix}$$

The quadratic performance indices are

$$J_i = \int_0^\infty (x^T Q_i x + u_1^T R_{i1} u_1 + u_2^T R_{i2} u_2 + u_3^T R_{i3} u_3)\ dt \quad (i = 1, 3)$$

The input weighting matrices are selected as follow

$$R_{11} = 1,\ R_{12} = 0,\ R_{13} = 0,$$

$$R_{21} = 0,\ R_{22} = 1,\ R_{23} = 0,$$

$$R_{31} = 0,\ R_{32} = 0,\ R_{33} = 1$$

Partition of the control task results in

$$B_1 = [0.0000 \quad -0.1200 \quad 0.0000 \quad 4.4190 \quad 1.5750]^T$$

$$B_2 = [0.0000 \quad 1.0000 \quad 0.0000 \quad 0.0000 \quad 0.0000]^T$$

$$B_3 = [0.0000 \quad 0.0000 \quad 0.0000 \quad -1.6650 \quad -0.0732]^T$$

In order to compare a game theoretic state feedback controllers with a LQR state feedback controller, the compatible performance index is chosen as

$$J = J_1 + J_2 + J_3 = \int_0^\infty (x^T Q x + u^T R u)\ dt.$$

where the state weighting matrix, Q, and the control weighting matrix, R, are

$$Q = Q_1 + Q_2 + Q_3, \quad R = diag([R_{11}\ R_{22}\ R_{33}])$$

The open-loop poles for the above system matrix of the aircraft model are located at $[-0.7801 \pm 1.0296j,\ -0.0176 \pm 0.1826j,\ 0.0000]$. These poles are associated

with the short period, phugoid period, and altitude, respectively. Assume that the control task is to move the last open-loop pole (located at 0.0000) of the system from its current location to a new location at -0.5 and to minimize the envelop bound of full-order state time response. The solution to LQR controller (Eq. (5.34)) is

$$P_r = 2.1037$$

In 3-player game, let a be a single pole of the linear system which is to be moved from its current location to its new location, say α. The associated equality constraint equation for a single pole shifting case is

$$\mu = \alpha - \sum_{i=1}^{3} B_{i_r} R_{ii}^{-1} B_{i_r}^T P_{i_r} \tag{6.25}$$

Based on the superimposed optimization criterion that minimizes the envelop bound of the full-order state response, the solutions to game theoretic controller (Eq. (6.25)) is

$$P_{1_r} = 3.8651, \quad P_{2_r} = 1.8779, \quad P_{3_r} = 48.4925$$

The 2-norm of N associated with both controller design are listed in Table 6.8. The results show the game theoretic controller has a smaller $\|N\|_2$ than the LQR controller. Hence, the state response for the closed-loop system using the game theoretic controller should have a tighter response envelop as compared to the LQR design.

The time response comparisons between the two controllers are shown in Fig. 6.7. The initial state conditions are $x_0 = [0,\ 1,\ 0,\ 0,\ 2]^T$. Figures 6.7(a), 6.7(b), 6.7(c), 6.7(d), and 6.7(e) represent correspondingly the altitude, forward speed, pitch angle, pitch rate and vertical speed state responses for the closed-loop system for both controllers. Figures 6.7(f), 6.7(g) and 6.5(h) show the spoiler, forward acceleration, and elevator responses.

Under the current perturbed configuration, the results show that the time response of all the state variables for the game theoretic controller have less overshoots than the one for the LQR design. Particularly, the overshoots associated with altitude, forward speed and pitch angle state responses for game theoretic controller are only 50% of the associated responses for the LQR controller. This results agree with the $\| N \|_2$ results in Table 6.8.

Table 6.8 Design Specification for State Feedback Controllers in Example 6.6

Controller	$\| N \|_2$	Superimposed Optimization Criteria
NASH	1.6633	Minimization of envelop bound of full-order state time response
LQR	2.1037	–

6.5 Summary

The inverse procedure proposed to solve differential game problems has a numerical advantage over the regular approach, since it does not require the solution of the complex coupled algebraic Riccati equations. Three optimization models are proposed as additional optimization criteria: (1) Minimum square of the Frobenius norm of the reduced-order state feedback gain matrix, (2) Minimize the envelop bound of reduced-order state time response, and (3) Minimize the envelop bound of full-order state time response.

The proposed optimization methodology not only performs both single pole and a pair of poles shifting, but also has the capacity to incorporate other (different from the chosen superimposed optimization criterion) controller design specifications.

A lateral motion model of an F-4 aircraft is used to demonstrate the procedure. Both unconstrained and constrained state feedback controller are considered for the minimal envelop bound of the full-order state time response. For the unconstrained case (i.e., no constraints on the individual entries of the state feedback gain matrix), the numerical results show that the closed-loop system using a game theoretic controller has less overshoot than the closed-loop system using a LQR controller. This is in agreement with the proposed optimization criterion. For the constrained case, the design task can only be implemented with a game theoretic controller.

A longitudinal motion model of an AIRC aircraft with three active controllers is used to demonstrate the extensile of the proposed procedure. The results show that this procedure can be extended to solve the pole placement problems for the general n-player games.

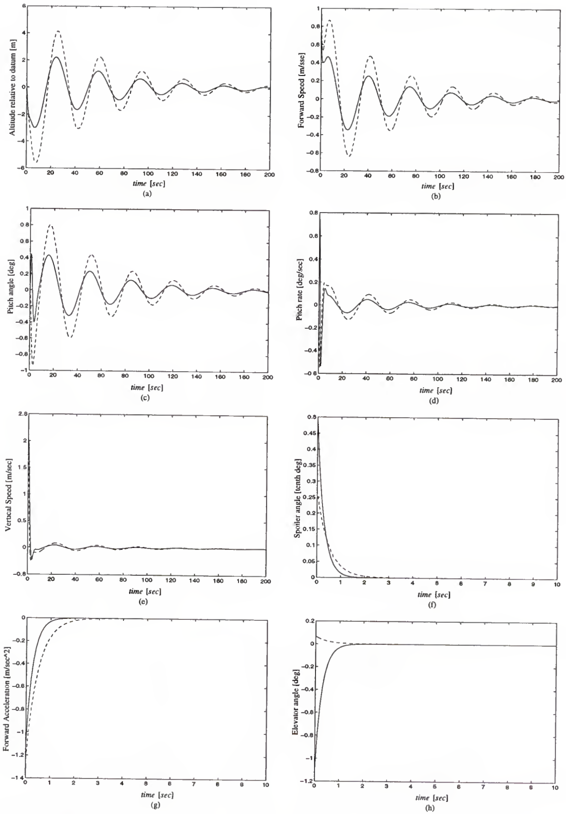

Figure 6.7 LQR controller vs. game theoretic controller (notation : solid line: game theoretic controller (minimization of full-order state response), dash line: LQR controller (minimization of full-order state response).

CHAPTER 7
SUMMARY

7.1 Summary of Results

In this research, two methodologies for designing a linear time invariant game theoretic state feedback controller in the regulator problem are proposed. First, given a set of state and input weighting matrices and their associated integral quadratic performance indices, integrative schemes based on Homotopy theory are utilized to solve the resulting coupled algebraic Riccati equations. To improve the efficiency of the algorithm, the symmetric property of Riccati matrices is explored in order to reduced the number of the unknowns to be solved. Second, given a set of desired closed-loop pole locations and the input weighting matrices in the integral quadratic performance indices, an inverse methodology for designing a linear time invarient full state feedback game theoretic controller is developed.

In the forward design process (i.e., integral quadratic performance indices and their associated input and state weighting matrices are given), the developed Homotopy algorithm for an algebraic Riccati equation has two advantage over existing Homotopy code:

(1) The developed algorithm determines the positive definite solution of an algebraic solution without the need of finding all the possible solutions.

(2) The developed algorithm is considered computationally efficient, since it utilized the symmetric property of the solution of an algebraic Riccati equation in conjunction with the Kronecker sum.

For coupled algebraic Riccati equations, the proposed algorithm is also very efficient computationally (in comparison with other homotopy algorithms). Numerical examples demonstrate that the proposed algorithm can solve problems that a Newton-iterative algorithm cannot solve.

In the inverse procedure, a new optimal pole placement methodology based on differential game theory for a general multiobjective linear regulator problem is proposed and developed. By using multiple integral quadratic performance indices, the control domain can be expanded and thus allow the control system designer the freedom of implementing additional criteria to specify the system's performance. The developed method is capable of shifting either a single pole or a pair of poles (two real poles or a pair of complex conjugate poles) to the desired closed-loop location(s) while satisfying additional design specifications for the linear system. Multiple poles shifting tasks can be implemented by utilizing the shifting algorithm from a single pole or a pair of poles recursively. It is found there are several advantages to the proposed methodology.

(1). The proposed methodology can determine state feedback controllers in an n-player differential game without actually solving n-coupled algebraic Riccati equation numerically,

(2). The proposed method can incorporate a design specification (e.g., minimize the envelop bound of full-order state time response) as an optimization criterion.

(3). The proposed methodology is better than conventional inverse LQR scheme.

(4). The proposed methodology can incorporate other controller design considerations (e.g., constraints on the elements of the state feedback gain matrices) as additional equality or inequality constraints for this optimization problem.

7.2 Suggestions for Future Work

Some of the results of this research are open-ended. For example, the Homotopy algorithm for coupled algebraic Riccati equations may not lead to positive definite solutions, although the author has not experienced any. Some areas where research can be invested with immediate benefit are outlined below.

(1). Modify the current Homotopy algorithm for the coupled algebraic Riccati equations to ensure the obtained solution is positive definite, which is the necessary

condition for the state feedback game theoretic controller to stabilize the closed-loop system.

(2). For the inverse procedure, other design specifications can be considered as superimposed optimization criteria for different dynamical systems (i.e., the set of criteria presented in Chapter 6 is not exhaustive).

(3). The proposed design of game theoretic controller is based on the Nash strategy, there are few researches on the other game theoretic strategies (i.e., Mini-Max, Stackelberg, Pareto strategies) has rarely found in literature, especially through the inverse procedure. Thus, it is of interest to conduct research on these strategies.

(4). The current research is for the deterministic models based on the time domain analysis, the stochastic process and/or frequency domain analysis could also be a rich area of research.

(5). In this research, the analytical model for a system is assumed exact. However, in a real control problem, there always exists the difference between an analytic model and a "true" model. Thus, the study on the issue of the "robust" pole placement technique for the multiobjective optimization problems should be carried on to make the current research results applicable to the real control problems.

APPENDIX: POLE PLACEMENT BY LINEAR QUADRATIC REGULATOR

The content of this appendix is based mainly on Arar and Sawan's work [59]. However, the notation is modified for the purpose of clarity. A correction of Arar and Sawan's paper on the optimization criterion for a pair of pole shifting is also made.

A1 Mathematical Model for Linear Quadratic Regulator

A linear controllable and observable dynamical system which is mathematically modelled in a state equation form is,

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{A1}$$

where $u(t) \in \Re^m$ represents control input to the plant, and $x(t) \in \Re^n$ represents the prefect measured plant state. The control law, u, are restricted to state feedback (i.e., $u(x,t)$). Matrices A and B are appropriately dimensioned. The performance index is defined as

$$J = \int_0^\infty (x^T Q x + u^T R u)\ dt \tag{A2}$$

where Q are positive semi-definite matrices, and R are positive definite matrices. The state feedback controllers, u, are

$$u(t) = -R^{-1}B^T P x(t) \tag{A3}$$

where P is the positive definite solution of the following algebraic Riccati equation

$$Q + PA + A^T P - PBR^{-1}B^T P = 0 \tag{A4}$$

Here, we assume that a set of closed-loop eigenvalues is given and a control weighting matrix, R, is pre-chosen. Arar and Swan's scheme [59] is used to find state weight matrices, Q, such that the system poles are shifted to the desired locations.

A2 Pole Placement by LQR Controller

A2.1 Preliminary Transformation

Consider the following transformation

$$x_r = T^T x. \tag{A5}$$

where $x \in \Re^n$ represents the plant states, $x_r \in \Re^r$ represents the "reduced" plant states, and $T \in \Re^{n \times r}$ are the reduced left eigenvectors of the open-loop matrix, A, associated with the open-loop poles which will be shifted. Apply this transformation to Eq. (A1) and yield the reduced system equation

$$\dot{x}_r(t) = A_r x_r(t) + B_r u_r(t). \tag{A6}$$

where $A_r \in \Re^{r \times r}$ and $B_r \in \Re^{r \times m}$ are

$$T^T A = A_r T^T. \tag{A7}$$

$$B_r = T^T B \tag{A8}$$

The reduced performance index associated with Eq. (A2) is

$$J_r = \int_0^\infty (x_r^T Q_r x_r + u_r^T R u_r)\ dt. \tag{A9}$$

where $Q_r \in \Re^{r \times r}$ is

$$Q = T Q_r T^T \tag{A10}$$

The reduced state feedback control law, u_r, of the reduced system is

$$u_r(t) = -R^{-1} B_r^T P_r x_r(t) \tag{A11}$$

P_r is the positive definite solution of the following reduced algebraic Riccati equation

$$Q_r + P_r A_r + A_r^T P_r - P_r B_r R^{-1} B_r^T P_r = 0 \tag{A12}$$

where

$$P = T P_r T^T. \tag{A13}$$

The reduced state feedback gain matrix is

$$K_r = R^{-1}B_rP_r. \tag{A14}$$

The reduced closed-loop system resulting from the reduced control law becomes

$$\dot{x}_r(t) = (A_r - B_rK_r)x_r(t). \tag{A15}$$

Substitution of Eqs. (A5), (A7), (A8), (A13) and (A14) into Eq. (A15), the full order model of the close-loop system becomes

$$\dot{x}(t) = (A - BK)x(t). \tag{A16}$$

where

$$K = K_rT^T = R^{-1}B_r^TP_rT^T \tag{A17}$$

A2.2 Shifting a Single Pole

Let a single pole, α, of the linear system be moved from a current location to a new location, say μ. Choose T to be the left eigenvector associated with the left eigenvalue, α, of the system, then the reduced system matrix in Eq. (A15) is

$$\mu = \alpha - B_rR^{-1}B_r^TP_r. \tag{A18}$$

The scalar P_r can be solved as

$$P_r = \frac{(\alpha - \mu)}{B_rR^{-1}B_r^T}. \tag{A19}$$

and the state weighting matrix that will shift a system pole form λ to μ is

$$Q_r = -2\lambda P_r + P_rB_rR^{-1}B_r^TP_r. \tag{A20}$$

A2.3 Shifting a Pair of Poles

Let a pair of the left eigenvalues, $\alpha + \beta j,\ \xi - \beta j$, of a $n \times n$ system matrix A, with associated non-zero left eigenvectors, $u \pm jv$ (or u and v for the two real poles shifting case), to be shifted from current location to a new location at $\gamma + \delta j, \phi - \delta j$, where

$$\xi = \alpha, \ \text{if}\ \beta \neq 0$$

$$\xi \neq \alpha, \ \text{if}\ \beta = 0$$

and

$$\phi = \gamma, \quad \text{if}\, \delta \neq 0$$

$$\phi \neq \gamma, \quad \text{if}\, \delta = 0$$

The transformation matrix, T, is chosen as

$$T = \begin{Bmatrix} u^T \\ v^T \end{Bmatrix}$$

Notice that the above transformation matrix is only for a pair of complex conjugate poles shifting case. For the case of two real poles shifting, the transformation matrix still has the same form as shown in the above equation. u and v, however, are the left eigenvectors of the system matrix corresponding to the two real eigenvalues.

From Eqs. (A5), (A14), (A15), apply the above transformation matrix to Eq. (A1) and obtain the reduced system equation

$$\dot{x}_r(t) = A_{rc}\, x_r(t)$$

where A_{rc} is the reduced closed-loop system matrix

$$A_{rc} = A_r - B_r R^{-1} B_r^T P_r. \tag{A21}$$

The quantities A_r and P_r are defined as

$$A_r = \begin{bmatrix} \alpha & \beta \\ -\beta & \xi \end{bmatrix} \tag{A22}$$

$$P_r = \begin{bmatrix} x & z \\ z & y \end{bmatrix} \tag{A23}$$

and the reduced closed-loop system matrix, A_{rc}. In Arar and Sawan's paper [59], the following assumptions are made: (1) $u \in \Re^2$ and (2) the control weighting matrix R is equal to $diag([1\ \ 1])$. Based on these assumptions,

$$B_r R^{-1} B_r^T = \begin{bmatrix} a & b \\ b & c \end{bmatrix} \tag{A24}$$

The reduced closed-loop system matrix, A_{rc}, can be further derived and it has a characteristic polynomial equation as follow

$$s^2 + m_1 s + m_2 = 0 \tag{A25}$$

where

$$m_1 = h_1 x + h_2 y + h_2 z + h_4 \tag{A26}$$

$$m_2 = f_1 z^2 - f_1 xy + f_2 x + f_3 y + f_4 z + f_5 \tag{A27}$$

with

$$h_1 = l_1,\ h_2 = l_2,\ h_3 = 2l_3,\ h_4 = -\alpha - \xi \tag{A28}$$

$$l_1 = a^2 + b^2,\ l_2 = c^2 + d^2,\ l_3 = ac + bd \tag{A29}$$

$$f_1 = l_3^2 - l_1 l_2,\ f_2 = \beta l_3 - \xi l_1,\ f_3 = -\beta l_3 - \alpha l_2 \tag{A30}$$

$$f_4 = \beta l_2 - \alpha l_3 - \xi l_3 - \beta l_1,\ f_5 = \beta^2 + \alpha\xi \tag{A31}$$

Since A_{rc} has a complex conjugate (left) eigenvalue pairs, $\gamma + \delta j, \phi - \delta j$, the characteristic equation of the reduced system matrix, A_{rc}, can be also written as

$$s^2 + n_1 s + n_2 = 0 \tag{A32}$$

where

$$n_1 = -2\gamma \quad (-\gamma - \phi,\ \text{if } \delta = 0)$$

$$n_2 = \gamma^2 + \delta^2 \quad (\gamma\phi,\ \text{if } \delta = 0)$$

By equating Eq. (A25) with Eq. (A32), we get

$$m_1 = n_1 \tag{A33}$$

$$m_2 = n_2 \tag{A34}$$

Equations (A33) and (A34) are the only equality constraint equations for this problem. However, we have three unknowns (i.e. x, y, and z) to be solved. Since the solutions of Eqs. (A33) and (A34) are not unique, we have to superimpose an optimization criteria on this problem.

A3 Correction for the Ill-defined Optimization Model

In Reference 59, Arar and Sawan proposed an constraint optimization model (L1) for the case of shifting of a pair of pole as follow

Min ($trace\ P_r$)

subject to

$$\text{Eqs. (A33) and (A34)}$$

$$P_r > 0, \text{(i.e., } x > 0 \text{ and } xy - z^2 > 0)$$

❑

The Largrangian for the optimization model (L1) is

$$L = x + y + \lambda_{11}(h_1x + h_2y + h_3z + h_4 - n_1) +$$

$$\lambda_2(f_1z^2 - f_1xy + f_2x + f_3y + f_4z + f_5 - n_2) \tag{A35}$$

The gradient of L will give the first order necessary condition as

$$1 + \lambda_1h_1 + \lambda_2(f_2 - f_1y) = 0$$

$$1 + \lambda_1h_2 + \lambda_2(f_3 - f_1x) = 0$$

$$\lambda_1h_3 + \lambda_2(2f_1z + f_4) = 0 \tag{A36}$$

The Hessian matrix of L is

$$H(L) = \begin{bmatrix} 0 & -\lambda_2f_1 & 0 \\ -\lambda_2f_1 & 0 & 0 \\ 0 & 0 & 2\lambda_2f_1 \end{bmatrix} \tag{A37}$$

which has eigenvalues of $[-\lambda_2f_1,\ \lambda_2f_1,\ 2\lambda_2f_1]$. Recall that the second order necessary condition for any constraint optimization problem is that the Hessian matrix of the Largrangian is positive semi-definite. Clearly, the Hessian matrix in Eq. (A37) is positive semi-definite if and only if $f_1 = 0$, since λ_2 will not be zero. From Eqs. (A29) and (A30), we get

$$f_1 = -(ad - bc)^2 \tag{A38}$$

This implies that f_1 can be zero if only the determinant of B_r is zero. Hence, if the determinant of B_r is not zero, which is true for the most general case, the numerical solution for the optimization model (L1) is not a relative minimum.

Correction

There are many design criteria for a linear control system, e.g., the criterion stated in Sec. 6.1 can be used as a superimposed optimization criterion. Here, the author use a

different criterion, which is "minimum of the square of Frobenius norm of solution to the reduced algebraic Riccati equation." The reason for choosing this as an additional optimization criterion, is to make the symbolic derivation feasible. Base on the new proposed criterion, a pair of poles shifting problem can be re-constructed as the following optimization model (L2)

$$\text{Min } \| P_r \|_F^2$$

subject to

Eqs. (A33) and (A34)

$$P_r > 0, \text{(i.e., } x > 0 \text{ and } xy - z^2 > 0)$$

□

The Largrangian, L_2, for the optimization model (L2) is

$$L_2 = x^2 + y^2 + 2z^2 + \lambda_{11}(h_1 x + h_2 y + h_3 z + h_4 - n_1) + \lambda_{12}(f_1 z^2 - f_1 xy + f_2 x + f_3 y + f_4 z + f_5 - n_2) \quad \text{(A39)}$$

The gradient of L_2 will give the first order necessary condition as

$$2x + \lambda_{11}h_1 + \lambda_{12}(f_2 - f_1 y) = 0$$
$$2y + \lambda_{11}h_2 + \lambda_{12}(f_3 - f_1 x) = 0$$
$$4z + \lambda_{11}h_3 + \lambda_{12}(2f_1 z + f_4) = 0 \quad \text{(A40)}$$

The Hessian matrix of L_2 is

$$H(L_2) = \begin{bmatrix} 2 & -\lambda_{12}f_1 & 0 \\ -\lambda_{12}f_1 & 2 & 0 \\ 0 & 0 & 4 + 2\lambda_{12}f_1 \end{bmatrix} \quad \text{(A41)}$$

which has the eigenvalues of $\left[2 - \lambda_{12}f_1,\ 2 + \lambda_{12}f_1,\ 4 + 2\lambda_{12}f_1\right]$. Hence, the second order necessary condition is

$$-2 \le \lambda_{12}f_1 \le 2$$

REFERENCE

[1]. Fitz-Coy, N., and Jang, J.-W., "Conflict Reduction in Multi–Tasking Space Platforms," in Proceedings of the 1st Microdynamics and Accurate Control Symposium, Nice, France, Nov., 1992.

[2]. Flugge-Lotz, I., and Marbach, H., "The Optimal Control of Some Attitude Control Systems for Different Performance Criteria," Journal of Basic Engineering, ASME Transactions, Series D, Vol. 85, No. 2, 1963, pp. 165-176.

[3]. Isaacs, R., "Differential Games I, II, III, IV," RAND Corporation Research Memorandum RM-1391, 1399, 1411, 1468, 1954-1956.

[4]. Starr, A. W., and Ho, Y. C., "Nonzero–Sum Differential Games," Journal of Optimization Theory and Applications, Vol. 3, No. 3, 1969, pp. 184-206.

[5]. Krikelis, N. J., and Rekasius, Z. V., "On the Solution of Optimal Linear Control Problems under Conflict of Interest," IEEE Transactions on Automatic Control, Vol. AC-16, No. 2, April 1971, pp. 140-147.

[6]. Fitz-Coy, N. G., and Jang, J.-W., "Active Vibration Isolation of a Space Platform: A Game Theoretic Approach," Paper AAS 93-647, presented at AAS/AIAA Astrodynamics Specialist Conference, Victoria, B.C., Aug. 16-19, 1993.

[7]. Innocenti, M., and Schmidt, D. K., "Quadratic Optimal Cooperative Control Synthesis with Flight Control Applications," J. of Guidance, Control, and Dynamics, Vol. 7, No. 2, March-April 1984, pp. 206-214.

[8]. Zangwill, W. I., and Garcia. C. B., Pathways to Solutions, Fixed Points, and Equilibria, Prentice-Hall, Englewood Cliffs, N.J., 1981.

[9]. Wang, S.-S., and Pan, C.-T., "Homotopy Method for Solving Algebraic Riccati Equations," IEE Proceedings, Vol. 136, Pt. D, No. 2, Mar. 1989, pp. 68-72.

[10]. Watson, L. T., Billups, S. C., and Morgan, A. P., "Algorithm 652 Hompack: a Suit of Codes for Globally Convergent Homotopy Algorithms," ACM Transactions on Mathematical Software, Vol. 13, No. 3, Sep. 1987, pp. 281-310.

[11]. Gembicki, F. W., "Performances and Sensitivity Optimization: a Vector Index Approach," PhD thesis, Case Western Reserve University, Jan., 1974.

[12]. Nash, J., "Two-Person Cooperative Games," Econometrica, Vol. 18, 1953, pp. 155-162.

[13]. Berkowitz, L. D., and Fleming, W. H., "On Differential Games with Integral Payoff," Annals of Math. Study, No. 39, Princeton University Press, Princeton, N.J., 1957, pp. 413-435.

[14]. Rao, S. S., "Game Theory Approach for Multiobjective Structural Optimization," Computers & Structures, Vol. 25, No. 1, 1987, pp. 119-127.

[15]. Salukvadze, M. E., Vector-Valued Optimization Problem in Control Theory, Academic Press, New York, 1979.

[16]. Skowronski, J., "Coordinated Control of Independent two Robot Arms on a Moving Platform attained via Differential Game," Paper AIAA-89-0584, presented at the 27th Aerospace Sciences Meeting, Reno, Nevada, Jan., 1989.

[17]. Bryson, A. E., Jr., and Ho, Y. C., Applied Optimal Control, Hemisphere Publishing Co., Washington, D.C., 1975.

[18]. Kuhn, H. W., and Tucker, A.W., "Nonlinear Programming," in Proceedings of the second Berkeley Symposium on Mathematical Statistics and Probability, J. Neyman (editor), University of California Press, Berkeley and Los Angeles, Calif., 1961, pp. 481-492.

[19]. Censor, Y., "Pareto Optimality in Multiobjective Problems," Applied Mathematics and Optimization, Vol. 4, 1977, pp. 41-59.

[20]. Zadeh, L. A., "Optimality and Nonscalar-valued Performance Criteria," IEEE Transactions on Automatic Control, Vol. AC-8, 1963, pp. 1.

[21]. Cunha, N. O. D., and Polak, E., "Constrained Minimization under Vector-valued Criteria in Finite Dimensional Spaces," Journal of Mathematical Analysis and Applications, Vol. 19, 1967, pp. 103-124.

[22]. Waltz, F. M., "An Engineering Approach: Hierarchical Optimization Criteria," IEEE Transactions on Automatic Control, Vol. AC-12, April, 1967, pp. 179-180.

[23]. Abou-Kandil, H., "Solution of N-Person Stackelberg Games with nearly Cooperating Leaders," International Journal of Control, Vol. 45, No. 3, 1987, pp. 1043-1050.

[24]. Chen, C. I., and Cruz, J. B. Jr., "Stackelberg Solution for Two-Person Games with Biased Information Patterns," IEEE Transactions on Automatic Control, Vol., AC-17, No. 6, Dec. 1972, pp. 791–797.

[25]. Simaan, M., and Cruz, J. B. Jr., "On the Stackelberg Strategy in Nonzero–Sum Games," Journal of Optimization Theory and Applications, Vol. 11, No. 5, 1973, pp. 533-555.

[26]. Basar, T., and Olsder, G. J., Dynamic Noncooperative Game, Academic Press Inc., New York, 1982.

[27]. Gao, J., Jakubowski, A., Klompstra, M. B., and Olsder, G. J., "Time-dependent Cooperation in Games," Differential Games and Applications, Springer-Verlag, Berlin, 1989.

[28]. Papavassilopoulos, G. P., Medanic, J. V., and Cruz, J. B., Jr., "On the Existence of Nash Strategies and Solutions to Coupled Riccati Equations in Linear-Quadratic Games," Journal of Optimization Theory and Applications, Vol. 28, No. 1, 1979, pp. 49-76.

[29]. Basar, T., Game Theory and H^{∞}-Optimal Control: the Continuous-time Case, Lecture Notes in Control and Information Sciences, Vol. 156, Springer-Verlag, Berlin, 1991, pp. 171-186.

[30]. Graham, A., Kronecker Products and Matrix Calculus with Applications, Halsted Press, New York, 1981.

[31]. Anderson, B. D. O., and Moore, J. B. Linear Optimal Control, Prentice-Hall, Englewood Cliffs, N. J., 1971.

[32]. Skelton, R. E., Dynamic Systems Control: Linear Systems Analysis and Synthesis, John Wiley & Sons, New York, 1988.

[33]. Kwakernaak, H. and Sivan, R., Linear Optimal Control Systems, John Wiley & Sons, New York, 1972.

[34]. Laub, A. J., "A Schur Method for Solving Algebraic Riccati Equations," IEEE Transactions on Automatic Control, Vol. AC-24, 1979, pp. 913-921.

[35]. Ramesh, A. V., and Utku, S., "Computational Complexities and Storage Requirements," Journal of Guidance, Control, and Dynamics, Vol. 12, July-August, 1989, pp. 469-479.

[36]. Richter, S., Hodel, A. S., and Pruett, P. G., "Homotopy Methods for the Solution of General Modified Algebraic Riccati Equations," IEE Proceedings-D, Vol. 140, No. 6, November, 1993, pp. 449-454.

[37]. Richter, S., and Collins, E. G. Jr., "A Homotopy Algorithm for Reduced Order Compensator Design Using the Optimal Projection Equations," Proceedings of 28th IEEE Conference on Decision and Control, Vol. 1, 1989, pp. 506-511.

[38]. Bernstein, D. S., and Hyland, D. C., "The Optimal Projection/Maximum Entropy Approach to Design Low-order, Robust Controllers for Flexible Structures," Proceedings of 24th IEEE Conference on Decision and Control, Vol. 2, 1985, pp. 745-742.

[39]. Veillette, R. J., Medanic, J. V., and Perkins, W. R., "Robust Stabilization and Disturbance Rejection for Systems with Structured Uncertainty," Proceedings of 28th IEEE Conference on Decision and Control, Vol. 1, 1989, pp. 936-941.

[40]. Jamshidi, M, and Böttiger, F, "On the Imbedded Solution of Algebraic Matrix Riccati Equation," Proceedings of Joint Automatic Control Conference, American Society of Mechanical Engineers, New York, 1976, pp. 474-481.

[41]. Collins, E. G., Jr., Davis, L. D., and Richter, S., "Homotopy Algorithm for Maximum Entropy Design," Journal of Guidance, Control, and Dynamics, Vol. 17, No. 2, March-April, 1994, pp. 311-321.

[42]. Jang, J.-W., and Fitz-Coy, N. G., "An Efficient and Robust Algorithm for the Algebraic Riccati Equation," 1995 SIAM Annual Meeting, Charlotte, North Carolina, Oct. 23-26, 1995.

[43]. Graham, A. M. A., Kronecker Products and Matrix Calculus: with Application, John Wiley & Sons, New York, 1981.

[44]. Gopal, M., Modern Control System Theory, John Wiley & Sons, New York, 1993.

[45]. Wiberg, D. M., State Space and Linear Systems, McGram-Hill Book Company, New York, 1971.

[46]. Gantmacher, F. R., Applications of the Theory of Matrices, Interscience Publishers, Inc., New York, 1959.

[47]. Molinari, B. P., "The Time-Invariant Linear-Quadratic Optimal Control Problem," Automatica, Vol. 13, 1977, pp. 347-357.

[48]. Strikant, R., and Basar, T., "Iterative Computation of Noncooperative Equilibria in Nonzero-sum Differential Games with Weakly Coupled Players," Proceedings of the 29th Conference on Decision and Control, Vol. 1, 1990, pp. 252-257.

[49]. Fitz-Coy, N. G., and Jang, J.-W., "A Game Theoretic Approach for Active Vibration Isolation of a Space Platform:," to be published by Journal of British Interplanetary Society in 1996.

[50]. Collins, E. G., Jr., Haddad, W. M., and Ying, S. S., "Reduced-order Dynamics Compensation Using the Hyland-Bernstein Optimal Projection Equations," to be submitted for publication, 1995.

[51]. Bass, R. W., and Gura, I., "High Order Design via State-Space Considerations," Proceedings of the 1965 Joint Automatic Control Conference, Troy, N.Y., 1965, pp. 311-318.

[52]. Ackermann, J., "Der Entwurf Linearer Regelungssysteme im Zustandsraum," Regulungestechnik und Prozessedatenverarbeitung, Vol. 7, 1972, pp. 297-300.

[53]. Andry, A. N., Jr., Shapiro, E. Y., and Chung, J. C., "Eigenstructure Assignment for Linear Systems," IEEE Transactions on Aerospace and Electronic Systems, Vol. AES-19, No. 5, Sep. 1983, pp. 711-728.

[54]. Kalman, B. E., "When Is a Linear Control System Optimal," ASME Journal Basic Engineering, Ser. D., Vol. 86, 1964, pp. 51-60.

[55]. Solheim, O. A., "Design of Optimal Control Systems with Prescribed Eigenvalues," International Journal of Control, Vol. 15, No. 1, 1972, pp. 143-160.

[56]. Graupe, D, "Derivation of Weighting Matrices towards Satisfying Eigenvalue Requirements," International Journal of Control, Vol. 16, No. 5, 1972, pp. 881-888.

[57]. Amin, M. H., "Optimal Pole Shifting for Continuous Multivariable Linear Systems," International Journal of Control, Vol. 41, No. 3, 1985, pp. 701-707.

[58]. Saif, M., "Optimal Linear Regulator Pole-placement by Weight Selection," International Journal of Control, Vol. 50, No. 1, 1989, pp. 399-414.

[59]. Arar, A.-R., and Sawan, M. E., "Optimal Pole Placement with Prescribed Eigenvalues for Continuous System," Journal of the Franklin Institute, Vol. 330, No. 5, 1993, pp. 985-994.

[60]. Jang, J.-W., and Fitz-Coy, N. G., "Design of Game Theoretic Controllers with Prescribed Eigenvalues," to be appeared in AASRC 37th Annual Symposium, Taipei, Taiwan, R.O.C., December 17, 1995.

[61]. Rao, S. V., and Lamba, S. S., "Eigenvalue Assignment in Linear Optimal-control Systems via Reduced-order models," Proceedings IEE, Vol. 122, No. 2, Feb. 1975, pp. 197-201.

[62]. Kreindler, E., and Rothschid, D., "Model Following in Linear Quadratic Optimization," AIAA Journal, Vol. 14, 1976, pp. 835-842.

[63]. Van Loan, C., "The Sensitivity of the Matrix Exponential," SIAM Journal on Numerical Analysis, Vol. 14, No. 6, Dec. 1977, pp. 971-981.

[64]. Golub, G. and Van Loan, C., Matrix Computations, John Hopkins University Press, Baltimore, 1989.

[65]. Onoda, J, and Haftka, R. T., "An Approach to Structure/Control Simultaneous Optimization for Large Flexible Spacecraft," AIAA Journal, Vol. 25, No. 8, Aug. 1987, pp. 1133-1138.

[66]. Luenberger, D. G., Linear and Non-linear Programming, Addison-Wesley Publishing Co., Reading, Mass., 1984.

[67]. Grace, A., Optimization Toolbox User's Guide, The MathWorks, Inc., Natick, Mass., 1992.

[68]. Fletcher, R., "A New Approach to Variable Metric Algorithms," Computer Journal, Vol. 13, pp. 317-322, 1970.

[69]. Davidon, W. C., "Variable Metric Method for Minimization," A.E.C. Research and Development Report, ANL-5990, 1959.

[70]. Bernstein, D., and So, W., "Some Explicit Formulas for the Matrix Exponential," IEEE Transactions on Automatic Control, Vol. 38, No. 8, Aug. 1993, pp. 1228-1232.

[71]. Hung, Y. S., and MacFarlane, A. G. J., Multivariable Feedback: A quasi-classical Approach. Lecture Notes in Control and Information Sciences, Vol. 40, Springer-Verlag, Berlin, 1982.

BIOGRAPHICAL SKETCH

Jiann-Woei Jang was born in Taipei, Taiwan, on September 20, 1963. In June, 1986, he received the Bachelor of Aeronautical Engineering degree from the Tamkung University, Tamsui, Taiwan. After a two-year military service in the air force from his home country, he worked as an engineer in the Tai-Engineering Co., which was the sole-sale agent of the German Siemens AG.

In August, 1989, he and his wife, De-Hui Hsu, became graduate students at the University of Florida. In May, 1991, he obtained a Master of Science degree in aerospace engineering. His wife received the degree of Master of Science in Nursing from the Nursing school in December of 1991.

In January, 1992, he became a graduate teaching assistant at the University of Florida; later that year in August, he also became a graduate research assistant. On January 10, 1993, a son, Alexander Jang, was born to De-Hui Hsu and Mr. Jang. In spring 1995, he received the outstanding teaching assistant of the year award. He is currently completing the requirements for the degree of Doctor of Philosophy in the Department of Aerospace Engineering, Mechanics, and Engineering Science.

Jiann-Woei Jang is also a student member of the American Institute of Aeronautics and Astronautics.

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

N. Fitz-Coy

Norman Fitz-Coy, Chairman
Assistant Professor of Aerospace Engineering, Mechanics, and Engineering Science

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Ulrich H. Kurzweg

Ulrich Kurzweg
Professor of Aerospace Engineering Mechanics, and Engineering Science

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Robert J. Hirko

Robert Hirko
Associate Engineer of Aerospace Engineering, Mechanics, and Engineering Science

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Bernard Mair

Bernard Mair
Associate Professor of Mathematics

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Haniph Latchman
Associate Professor of Electrical and Computer Engineering

This dissertation was submitted to the Graduate Faculty of the College of Engineering and to the Graduate School and was accepted as partial fulfillment of the requirements for the degree of Doctor of Philosophy.

December 1995

Winfred M. Phillips
Dean, College of Engineering

Karen A. Holbrook
Dean, Graduate School